REȚELE DE CALCULATOARE.
Proiectare şi administrare

Partea I
Arhitecturi şi standarde

Capitolul 1: Modele şi standarde

1.1 A fost odată

Cîte calculatoare conţine INTERNETUL? Se aproximează ca ar fi peste 40 de milioane, dar nimeni nu are curaj să avanseze o cifră exactă. Popularitatea Internetului se datorează în principal World Wide Web-ului şi poştei electronice.

Poştă electronică, ştiri, biblioteci, muzică şi multe altele. Pe *net* găseşti de toate. Este unul din motivele pentru care cei din branşă au ajuns să fie numiţi *netocraţi* după o evoluţie care a durat vreo 40 de ani. Cîţi dintre noi mai pot trăi astăzi fără porţia zilnică de Internet?

Toate lucrurile rele au şi o parte bună. De ce avem curajul să afirmăm aşa ceva? Pentru că dacă în 1957 URSS nu ar fi plasat pe orbita Pamîntului primul satelit artificial (Sputnik) astăzi nu am fi avut probabil Internet. Ca răspuns la această palmă primită, în cadrul Departamentului de Apărare al SUA (DoD) ia fiinţă *Advanced Research Projects Agency (ARPA)*. Americanii doreau să recupereze handicapul şi să devină lideri mondiali în domeniul ştiinţelor şi tehnologiilor cu aplicaţii militare. Cinci ani mai tîrziu (1962), *Paul Baran* de la Rand Corporation a fost însărcinat de către US Air Force să întocmească un studiu cu privire la posibilitatea menţinerii comenzilor şi controlului asupra rachetelor intercontinentale în cazul unui atac nuclear din partea URSS. Trebuia dezvoltată o reţea informatică descentralizată astfel încît orice oraş american ar fi fost atacat, armata să poată declanşa un contraatac nuclear.

Baran descrie mai multe variante prin care se putea ajunge la rezultatul scontat iar în final face o propunere concretă: comutarea pachetelor presupune descompunerea datelor în pachete (sau datagrame) care vor conţine informaţii cu privire la originea şi detinaţia lor. Aceste pachete vor circula prin reţea pînă cînd ajung la destinaţie.

Sfîrşitul anului 1968, începutul lui 1969 marchează naşterea Internetului. În 1968, National Physical Laboratory din Marea Britanie realizează şi testează prima reţea construită după principiile lui Baran. La scurt timp, Advanced Research Projects Agency din cadrul Pentagonului decide realizarea unei reţele mai mari. În toamna anului 1969 la UCLA se realizează primul nod al acestei reţele iar pînă în decembrie mai apar încă 3: University of California (Los Angeles), SRI (Stanford), University of California (Santa Barbara) şi University of Utah.. Această reţea va fi denumită ARPANET după numele sponsorului agenţiei Pentagonului. Cele patru noduri puteau transmite informaţii între ele

prin intermediul unor linii dedicate şi chiar puteau fi programate de la distaţă. Pînă în 1972 ARPANET va ajunge la 35 de noduri principalul trafic al reţelei constînd în ştiri şi mesaje personale.

În acelaşi an, *Ray Tomlison* de la BBN realizează primul program de poştă electronică. Interesant este că acesta a fost folosit în principal pentru discuţii între împătimiţii de science fiction (lista se numea SF-Lovers).

Standardul original pe care se bazau comunicaţiile reţelei ARPA s-a numit NCP – Network Control Protocol. Însă pe măsură ce anii au trecut şi tehnica a avansat *Vinton Cerf* de la Stanford împreună cu *Bob Kahn* de la DARPA (noua denumire a reţelei ARPA) au înlocuit în 1973 NCP cu TCP/IP – Transmission Control Protocol/Internet Protocol. Acest nou protocol oferea suport pentru comunicaţie pentru calculatoare diferite. Mai mult, TCP/IP a fost folosit pentru a lega între ele calculatoare care nu făceau parte din ARPANET (în 1983 reţeaua militară s-a desprins sub denumirea MILNet)

Anul 1976 este unul al relizărilor deosebite. Dr. *Robert M. Metcalfe* dezvoltă tehnologia Ethernet care perimite transferul de date pe cablu coaxial. A fost o realizare crucială pentru domeniul reţelelor. În acelaşi an ia naştere SATNET reţeaua care lega SUA de Europa. Chiar dacă pare surprinzător această reţea folosea sateliţii INTELSAT aflaţi în proprietatea unui consorţiu de ţări europene.

Trei ani mai tîrziu, un tînăr absolvent al Univeristăţii North Carolina, Steve Bellovin, împreună cu programatorii Tom Truscott şi Jim Ellis dezvoltă USENET. Era prima reţea descentralizată dedicată în exclusivitate ştirilor. Această reţea se baza pe protocolul UUCP (Unix-to-Unix CoPy) dezvoltat de AT&T Bell Labs şi distribuit împreună cu sistemul de operare Unix.

Similar cu USENET, reţeaua BITNET (Because It's Time Network) conecta maincadruruile IBM din mediul educaţional pentru a oferi serviciii de poştă electronică

Despre anul 1983 putem spune că este anul Internet: se înfiinţează IAB – Internet Activities Board. Începînd cu prima zi a acestui an toate calculatoarele conectate la ARPANET folosesc TCP/IP şi renunţă la NCP. În sprijinul IAB a venit Univeristatea din Wisconsin carea a creat Domain Name System –DNS. Acesta permitea pachetelor să fie direcţionate către un nume de domeniu unde i se aloca adresa IP corespunzătoare. Utilizatorii au putut astfel să acceseze mult mai uşor serverele reţelei.

Cercetările şi dezvoltările continuă pe parcursul următorilor ani. Astfel, autoritatea care dirijează evoluţia Internet - ului este **ISOC** (**Internet SOCiety**), înfiinţată în ianuarie 1992, cu scopul de a promova utilizarea Internet - ului şi de a prelua administrarea sa. În cadrul acestei organizaţii o parte din membrii sunt reuniţi în cadrul **IAB** care are responsabilitatea tehnică a evoluţiei reţelei şi defineşte standardele Web. Membrii acestui consiliu au întîlniri regulate în care sunt acceptate noi standarde, alocă adresele şi păstrează o listă a numelor care trebuie sa rămînă unice. Comunicările sunt puse la dispoziţie printr-o serie de rapoarte tehnice, numite **RFC** - uri (**Request For Comments**) care sunt memorate on-line şi pot fi citite de oricine este interesat de ele.

Consiliul IAB este format din patru grupuri principale, şi anume: **IRTF** (**Internet Research Task Force)** care are rolul de a rezolva problemele pe termen lung, **IETF** (**Internet Engineering Task Force**) care are sarcina de a rezolva problemele pe termen scurt, **IESG** (**Internet Engineering Steering Group**) şi **IRSG** (**Internet Research Steering Group**). Aceste grupuri sunt responsabile cu evaluarea şi testarea proiectelor şi a standardelor propuse, pentru a determina dacă o propunere merită să devină un standard Internet.

Pe tot parcursul anilor care s-au scurs dezvoltările au continuat. Comunicarea în Internet a devenit o sursă inepuizabilă şi rapidă de documentare, depăşind complet toate

celelalte surse de documentare. Internet-ul pune la dispoziția utilizatorilor (cel puțin) cinci servicii extrem de importante şi anume:

1.poşta electronică (e-mail): poate fi folosită pentru a expedia mesaje unor persoane din orice colț al lumii;

2.transferul fişierelor (ftp: File Transfer Protocol): este folosit la transferul fişierelor (text sau binare) între două calculatoare;

3.conectare la o maşină la distantă (telnet, ssh, Terminat Services): folosit la deschiderea unor sesiuni de lucru pe maşini aflate la distanță;

4.buletine de ştiri (usenet): utilizatorii se pot înscrie la grupuri de discuții;

5.Word Wide Web (sau WWW): facilitează transferul informației între servere Web.

Primele programe (browser-e, clienți de e-mail şi ftp) Internet aveau o interfață exclusiv bazată pe linia de comandă. Acest lucru a fost schimbat de World Wide Web, care oferă o prezentare grafică a Internet-ului fără a mai impune cunoaşterea unor comenzi cu sintaxa mai mult sau mai putin complicată. Deşi utilizează Internet-ul pentru funcționare, Web-ul se bazează pe un concept total diferit: hypertext-ul. Diferența între un text static (un articol de ziar) şi hypertext este uriaşă: în cazul hypertext-ului, calculatorul poate aduce direct pe ecran toate fragmentele la care textul curent are legături (cu doar un click!), fără a strica impresia de document unitar. Word Wide Web oferă hypertext şi, mai mult, depăşeşte simpla interconectare a documentelor text. Caracteristicile Web (JavaScript, CGI, etc) permit paginilor individuale de Web să interacționeze cu utilizatorul, iar serverele Web pot crea pagini Web bazate pe informații dinamice.

WWW este, astăzi, cel mai "vizibil" instrument Internet. Web-ul prezintă informația sub forma de text şi grafică şi a oferit afacerilor cea mai motivată rațiune pentru conectarea la Internet: usurința şi rapiditatea comunicațiilor.

Am început cu evoluția rețelelor prin prisma Internetului dintr-un singur motiv: în cele mai multe cazuri, cei din afara branşei asociază rețelele cu Internetul. Şi nu cred că greşesc.

Pînă la apariția Internetului, evoluția rețelelor a fost la fel de interesantă.

În faza lor incipientă rețelele nu se bazau pe o legătură fizică între calculatoare (*Sport-Shoe Networking)*. Mediul de comunicație era discheta, iar viteza de transmisie era determinată de viteza de deplasare a utilizatorului de la un calculator la altul (de unde şi numele). Astăzi am ajuns să discutăm de rețele pe fibră optică sau de cele fără fire. Întrebarea firească este *cum s-a ajuns aici?*

Firmele care dețineau calculatoare au fost nevoite să găsească o soluție prin care să-şi poată partaja resursele, să obțină o fiabilitate ridicată prin accesul la echipamente şi să economisească bani. Toate aceste probleme au fost rezolvate prin dezvoltarea tehnologiilor rețelelor.

Dacă în anii 70-80 cele mai multe companii puneau la dispoziția angajaților terminale conectate la calculatoare puternice, popularitatea rețelelor a crescut în momentul în care rețelele formate din calculatoare personale au oferit un raport preț/performanță optim. Începînd cu anii 90 accesul de la distanță la informații, comunicațiile interpersonale şi divertismentul interactiv au contribuit decisiv la expansiunea rețelelor. La acestea se adaugă sistemele de operare diponibile comercial.

1.2 Definirea rețelelor

În termeni foarte simpli, o *rețea* reprezintă un sistem de oameni şi obiecte conectate între ele. Oriunde privim în jurul nostru putem observa un anumit tip de rețea (sistemul nervos, sistemul cardiovascular etc). Rețelele de comunicații sunt proiectate astfel încît două calculatoare, localizate oriunde în lume, să fie capabile să comunice între ele, indiferent de tipul acestora (PC, Mac, maincadru etc). Acest lucru este posibil prin intermediul unei limbi comune, numită *protocol.*

Protocolul este definit ca un *set formal de reguli şi convenții cu ajutorul cărora este guvernat schimbul de informații între echipamentele unei rețele.*

Majoritatea rețelelor sînt clasificate în LAN (localizate de obicei într-o clădire, campus) sau WAN (acoperă o arie geografică mai mare). Ele sunt rezultatul dezvoltării aplicațiilor pentru mediul afacerilor. Dar fiecare din aceste aplicații aveau o manieră proprie de operare, independentă de calculator. Eficiența dispărea în acest caz. Organizațiile aveau nevoie de o soluție care să rezolve următoarele probleme:

- cum să fie evitată duplicarea echipamentelor?
- cum se poate comunica eficient?
- cum se poate gestiona o rețea?

O primă soluție la problemele organizației a fost realizarea rețelelor locale (LAN). Deoarece puteau conecta toate stațiile de lucru şi perifericele dintr-o clădire, LAN-urile au făcut posibilă utilizarea eficientă a tehnologiilor informaționale.

Însă pe măsură ce calculatoarele au început să fie folosite pe scară tot mai largă în domeniul economic, LAN-urile nu mai erau suficiente. A apărut necesitatea schimbului de informații între organizații, iar soluția a reprezentat-o apariția WAN

La început, dezvoltarea LAN şi WAN a fost haotică. Anii ’80 pot fi caracterizați printr-o expansiune extraordinară a rețelelor. Companiile au început să fie conştiente de economiile rezultate în urma folosirii tehnologiei rețelelor.

Dar la mijlocul anilor ’80, această creştere s-a oprit brusc. Multe din tehnologiile folosite în rețele se bazau pe soluții hard şi soft diferite, fapt care a dus la incompatibilitatea dintre acestea. Devenea din ce în ce mai dificilă comunicarea între rețelele care foloseau specificații diferite.

1.2.1 LAN

Rețele de tip LAN (Local Area Network) reprezintă cea mai comună implementare a tehnologiei rețelelor pentru firmele de mărime mică-medie. O astfel de rețea se carcaterizează prin:

- operare într-o arie geografică limitată;
- permite accesul utilizatorilor la medii de transmisie cu lățime de bandă mare;
- oferă conectivitate continuă pentru serviciile locale;
- conectează fizic echipamente adiacente.

Dacă privim acest tip de rețele prin prisma sistemelor de operare ce pot fi folosite vom putea clasifica aceste rețele în *peer-to-peer* şi *rețele bazate pe servere.*

Rețelele peer-to-peer nu implică existența unui server care să asigure serviciile în rețea. Fiecare din calculatoarele rețelei poate îndeplini şi funcția de server. Utilizatorul este cel care hotărăşte ce periferic sau informație doreşte să fie accesată şi de ceilalți membri ai rețelei. Acest tip de rețele se pretează cel mai bine pentru firmele mici unde datorită complexității relativ redusă a activităților nu este nevoi de servicii complexe.

La polul opus, rețelele bazate pe server(e) implică exsitența cel puțin a unui calculator care să joace rolul de server. În acest caz, informațiile care trebuie partajate în cadrul firmei vor fi gestionate de server. Aceasta este soluția adoptată de majoritatea firmelor şi ca urmare a posibilității gestionării centralizate a securității rețelei. Serverele dintr-o astfel de rețea pot îndeplini următoarele roluri:

- *servere de fişiere şi imprimare.* Aceste servere oferă un suport sigur pentru toate datele companiei şi gestionează tipărirea la imprimantele partajate în rețea.
- *servere pentru aplicații.* Sînt serverele care asigură componenta cu acelaşi nume pentru aplicațiile client-server. Exemple de astfel de servere: web-serverele, serverele pentru baze de date.
- *servere de mail.* Sînt serverele care gestionează mesajele electronice pentru clienții unei rețele.
- *servere pentru gestiunea securității.* Sînt serverele care asigură securitatea unei rețele locale cînd aceasta este conectată la o rețea de tipul Internetului. Pot fi incluse în această categorie firewall-urile, proxy serverele.
- *servere pentru comunicații.* Acestea sînt serverele care asigură shimbul de informații între între rețea şi clienții din afara acesteia (accesul prin dial-up de exemplu).

Pentru desemnarea manierei de proiectare a unei rețele se foloseşte termenul *topologie*. Trebuie să precizăm că există două tipuri de topologii: fizică şi logică. Topologia fizică a unei rețele se referă la configurația mediilor de transmisie, a calculatoarelor şi a perifericelor. Topologia logică reprezintă metoda folosită pentru transferul informațiilor de la un calculator la altul. Despre aceste topologiivom discuta pe larg într-un capitol viitor. Pentru moment facem o scurtă trecere în revistă a topologiilor fizice:

- bus (magistrală)
- star (stea)
- ring (inel)
- tree (arbore)

Se cuvine săf acem o mică observație: această clasificare nu are caracter absolut. Sînt lucrări în care topologia rețelelor este prezentată în altă manieră.

Rețele Bus[1]

Topologia magistrală este cea mai simplă modalitate de conectare a calculatoarelor într-o rețea: un singur mediu de transmisie (cablu) la care se conectează toate calculatoarele şi perifericele, denumite în continuare *noduri* (Fig. 1.2.1).

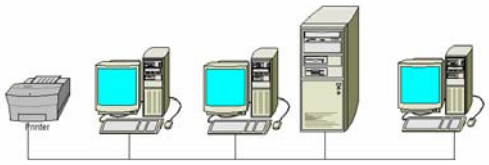


Fig. nr. 1.1.1 Topologie de rețea bus

[1] A nu se confunda cu *system bus* ul calculatorului.

Fiecare calculator verifică dacă informațiile pe care le primeşte îi sînt adresate sau nu. Despre acest mecanism vom discuta pe larg într-un capitol viitor.

Avantaje	Dezavantaje
Uşurință în conectarea calculatoarelor Necesarul de cablu este redus	Rețeaua nu funcționează dacă apar întreruperi în cablu Este nevoie de terminatori la ambele capete ale cablului Problemele sînt greu de identificat dacă rețeaua cade

Rețele Star

Într-o astfel de rețea fiecare nod este conectat direct la un *hub* sau *concentrator*.

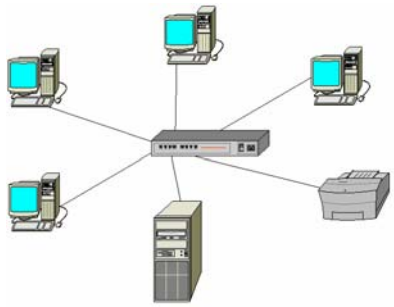


Fig. nr. 1.1.2: Topologie de rețea star

Informațiile sînt transmise de la calculatorul sursă către cel destinație prin intermediul hubului. Acesta este principalul dispozitiv care gestionează si controlează funcțiile rețelei.

Avantaje	Dezavantaje
Uşurință în instalare Rețeaua nu este afectată dacă sînt adăugate sau retrase calculatoare Usurință în detectarea prblemelor	Necesită cablu mai mult Sînt mai costisitoare Dacă un hub se defectează, toate calculatoarele din acel nod devin nefuncționabile

Rețele Ring

Din exterior, o astfel de rețea seamănă foarte mult cu o rețea star. Din punct de vedere al topolgiei logice însă, MAU (Multistation Access Unit) este dispozitivul care permite informațiilor să treacă de la un nod la altul în cadrul unui inel comunicațional.

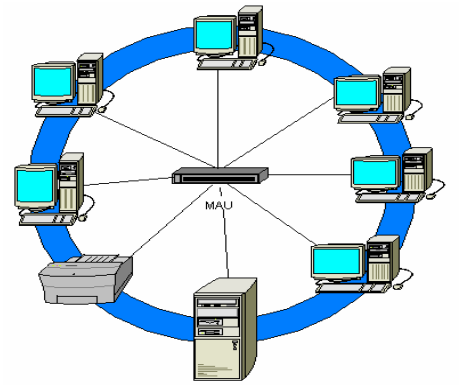


Fig. nr. 1.1.3: Topologie de rețea inel

Rețele Tree

Topologia de rețea tree combină caracteristicile topologiilor bus şi star. Nodurile sîn grupate în mai multe topologii star care la rîndul lor sînt legate la un cablu central. Acestea pot fi considerate topologiile ci cea mai bună scalabilitate.

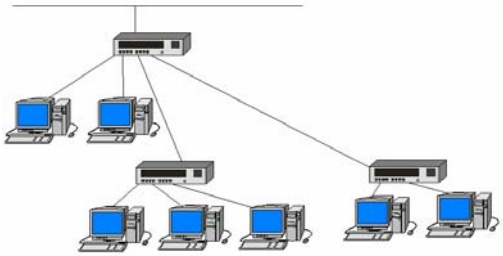


Fig. nr. 1.1.4: Topologie de rețea arbore

Avantaje	Dezavantaje
Segmentele individuale au legături directe	Lungimea maximă a unui segement este limitată Dacă apar probleme pe conexiunea principală sînt afectate toate calculatorele de pe acel segment

1.2.2 WAN

Pe măsură ce utilizarea calculatoarelor în domeniul economic a devenit omniprezentă, s-a ajuns la concluzia că LAN-urile nu mai corespundeau nevoilor firmelor.

Într-o rețea LAN, fiecare departament era privit ca o “insulă electronică”. A apărut însă necesitatea schimbului de informații între aceste insule iar soluția a reprezentat-o crearea

WAN-urilor (Wide Area Netowrk): rețele care interconectează LAN-uri, furnizind acces la calculatoare din alte locații geografice.

Tehnologiile folosite în cadrul WAN: modem-uri, ISDN (Integrated Services Digital Network), DSL (Digital Subscriber Loop), Cadru Relay, ATM (Asynchronous Transfer Mode) T-Carrier Series (în SUA T1, T2, T3), SONET (Synchronous Optical Network). Despre toate acestea vom discuta pe larg într-un capitol viitor.

1.3 Modelul de referință OSI

Conceptul de *nivel* este folosit pentru a vă ajuta să înțelegeți acțiunile şi procesele ce apar în timpul transmiterii informațiilor de la un calculator la altul.

Într-o rețea, comunicarea are la origine o sursă, apoi informația circulă pînă la o destinație. Informațiile care traversează rețeaua sînt referite ca *date*, *pachete* sau *pachete de date.*

Adresa sursă a unui pachet de date specifică identitatea calculatorului care transmite respectivul pachet. Adresa destinație precizează identitatea calculatorului care va recepționa respectivul pachet. Datele sînt grupate în unități logice de informații. Ele includ utilizatorul respectivelor informații şi alte elemente pe baza cărora este posibilă comunicarea.

Datele dintr-un calculator sînt reprezentate prin biti. Dacă un calculator ar transmite doar unul sau doi biți, nu ar fi o manieră prea eficientă de comunicare. Prin urmare, are loc o grupare a acestora în kilo, mega sau gigabytes.

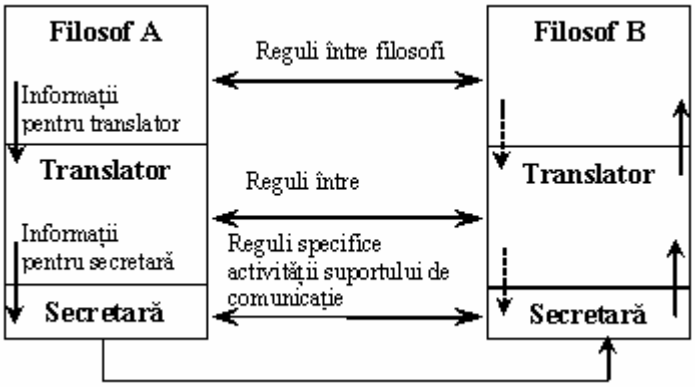
Am făcut deja referire la un alt element întîlnit în rețelele de calculatoare: "*mediul*". Acesta reprezintă un material prin care sînt transmise datele, şi poate fi unul din următoarele elemente:

- cablu telefonic
- cablu categoria 5 UTP
- cablu coaxial (cablu TV)
- fibră optică
- alte tipuri de cabluri bazate pe cupru

Mai există şi alte tipuri de media, dar acestea nu le vom lua în calcul în studiul rețelelor. În primul rînd este vorba de atmosfera prin care se propagă undele radio, microundele şi lumina. În al doilea rînd este vorba de undele electromagnetice care traversează Cosmosul, unde în mod virtual nu există molecule sau atomi. În aceste cazuri, comunicația este denumită *fără fir.*

Protocolul reprezintă un *set de reguli pe baza căruia se determină forma datelor şi transmisia acestora.* Exemplu al lui Andrew Tanenbaum[2] (comunicarea între doi filozofi) este un început bun pentru a înțelege ce presupune comunicarea bazată pe niveluri şi protocoale:

[2] Adaptare după Andrew S. Tanenbaum: "Retele de calculatoare", Editia a treia, Editura Agora, Tg. Mures, 1998, pg. 18



Nivelul *n* al unui calculator poate comunica cu nivelul *n* al altuia. Prin urmare se spune că regulile folosite în comunicare se numesc *protocoale de nivel n*.

Spuneam că dezvoltările timpurii din zona rețelelor au fost haotice, şi că începutul anilor ’80 se caracterizează printr-o expansiune a acestora. Singura modalitate prin care deținătorii de rețele puteau să “vorbească aceeaşi limbă” a fost agrearea din partea vînzătorilor şi producătorilor de echipamente de rețea a unui set comun de standarde.

International Organization for Standardization (*ISO*) este organizația care a cercetat şi dezvoltat scheme de rețele precum DECNET, SNA, TCP/IP. Rezultatul cercetărilor s-a concretizat într-un model de rețea care i-a ajutat pe producători să creeze echipamente compatibile între ele.

Modelul de referință OSI(Open Systems Interconnect), realizat în 1984, nu este altceva decît o schemă descriptivă care a pus la dispoziția vînzătorilor standardele necesare asigurării compatibilității şi interoperabilității între diferitele tehnologii. Şi este cel mai bun instrument pentru învățare.

Modelul de referință OSI, este primul model pentru standardizarea comunicațiilor în rețele. Există şi alte modele, dar majoritatea producătorilor de echipamente respectă aceste standarde. Acest model permite utilizatorilor să vadă funcțiile rețelei pe măsură ce ele apar la fiecare nivel în parte. Chiar dacă pare destul de abstract, este un instrument foarte bun pentru a ilustra modul în care informațiile traversează o rețea: explică, vizual, circulația datelor de la o aplicație, către mediul fizic de transmisie şi apoi către o altă aplicație localizată pe un calculator din rețea, chiar dacă expeditorul şi destinatarul fac parte din rețele cu topologii diferite.După cum se vede şi din figura alăturată, în modelul de referință OSI există 7 niveluri, fiecare dintre acestea ilustrînd o funcție particulară a rețelei. Separarea între funcțiile rețelei este denumită *nivelare (layering)*.

Modelul OSI este doar un model de arhitectură de rețea, deoarece spune numai ceea ce ar trebui să facă fiecare nivel, şi nu specifică serviciile şi protocoalele utilizate la fiecare nivel. Fiecare nivel al modelului OSI are un set predeterminat de funcții pe care le realizează pentru a duce la bun sfîrşit comunicarea.

Nivelul 7: Aplicație

Poetic vorbind, este nivelul situat cel mai aproape de inima utilizatorului. Prin ce diferă de celelalte niveluri ale modelului? Oferă servicii pentru aplicațiile utilizatorilor dar nu oferă servicii celorlalte niveluri.

Nivelul aplicaţie identifică şi stabileşte disponibilitatea partenerului de comunicaţie, sincronizează aplicaţiile între ele şi stabileşte procedurile pentru controlul integrităţii datelor şi erorilor. De asemenea identifică dacă există suficiente resurse pentru a sprijini comunicaţia între parteneri. Pentru a fi mai uşor să vă amintiţi despre acest nivel, gîndiţi-vă la browsere.

Nivelul 6: Prezentare

Este nivelul care asigură că informaţiile pe care nivelul aplicaţie al unui sistem le transmite, pot fi citite de către nivelul aplicaţie al altui sistem. Atunci cînd este necesar, nivelul aplicaţie face translaţie între diferitele formate ale datelor folosind un format comun pentru reprezentarea acestora. Trebuie să priviţi acest nivel ca cel la care are loc codificarea datelor în format ASCII, de exemplu.

Nivelul 5: Sesiune

După cum spune chiar numele său, acest nivel stabileşte, gestionează şi finalizează sesiunile de comunicaţie între aplicaţii. Prin sesiune se înţelege dialogul între două sau mai multe entităţi. Nivelul sesiune sincronizează dialogul între nivelurile sesiune ale entităţilor şi gestionează schimbul de date între acestea. În plus, acest nivel oferă garanţii în ceea ce priveşte expedierea datelor, clase de servicii şi raportarea erorilor. În cîteva cuvinte, acest nivel poate fi asemuit cu dialogul uman.

Nivelul 4: Transport

Este nivelul la care are loc segmentarea şi reasamblarea datelor. El furnizează un serviciu pentru transportul datelor către nivelurile superioare, şi în special caută să vadă cît de sigur este transportul prin reţea. Nivelul transport oferă mecanisme prin care stabileşte, întreţine şi ordonă închiderea circuitelor virtuale; detectează "căderea" unui transport şi dispune refacerea acestuia; controlează fluxul de date pentru a preveni rescrierea acestora. Gîndiţi-vă la calitatea servicilor sau la încredere!

Nivelul 3: Reţea

Este unul dintre cele mai complexe niveluri; asigură conectivitatea şi selecţia căilor de comunicaţie între două sisteme ce pot fi localizate în zone geografice diferite. Gîndiţi-vă la selectarea căilor de comunicaţie, switching, adresare şi rutare.

Nivelul 2: Legătură date

Este nivelul care asigură tranzitarea datelor de la nivelul fizic pe baza adresării fizice, topologiei reţelei, notificării erorilor, ordonarea cadru-urilor şi controlul fluxului informaţional. Gîndiţi-vă la cadru-uri şi controlul accesului.

Nivelul 1: Fizic

Defineşte specificaţiile electrice, mecanice, procedurale şi funcţionale necesare activării, întreţinerii şi dezactivării legăturii fizice între sisteme. Specificaţile vizează nivelul voltajului, ratele de transmisie a datelor, distanţa maximă de transmisie, conectorii fizici. Gîndiţi-vă la semnale şi medii de transmisii.

1.4 Modelul TCP/IP

Am început acest capitol cu modelul OSI deoarece reprezintă abecedarul acestui domeniu. Cu toate acestea, pentru transmisiile de date din cea mai mare rețea existentă – Internetul, standardul folosit este *TCP/IP (Transport Control Protocol/Internet Protocol)*. Acest model a fost creat de Ministerul Apararii din SUA care a dorit să construiască o rețea capabilă să reziste în orice condiții, chiar şi într-un război nuclear. Era extrem de important să fie creată o rețea capabilă să opereze cu o infrastructură distrusă în proporție de peste 90%, fără să aibă vreo importanță starea fizică a anumitor segmente ale rețelei.

Spre deosebire de OSI, modelul TCP/IP are doar patru niveluri: aplicație, transport, internet şi rețea. Deşi există două niveluri cu acelaşi nume ca la modelul OSI, nu trebuie confundate cu acelea pentru că fiecare nivel are funcții total diferite.

Nivelul aplicație

Proiectanții TCP/IP au considerat că protocoalele de nivel înalt din acest model trebuie să includă detalii cu privire la sesiunile de lucru şi modul de prezentare a datelor. Astfel, într-un singur nivel sînt combinate toate facilitățile legate de reprezentarea datelor, codificare şi controlul dialogului.

Nivelul transport

Acest nivel vizează calitatea serviciilor oferite: încrederea în transmisie, controlul fluxului de date şi corectarea erorilor. Unul din protocoalele întîlnite la acest nivel (Transport Control Protocol), oferă o modalitate flexibilă de realizare a comunicațiilor în rețea. Fiind un protocol orientat conexiune, dialogul dintre sursă şi destinație se realizează prin împachetarea informațiilor de la acest nivel în segmente.

Orientarea către conexiune nu înseamnă că între calculatoarele care comunică există vreun circuit, ci că segementele nivelului 4 circulă înainte şi înapoi între cele două calculatoare într-o perioadă de timp dată.

Nivelul internet

Scopul acestui nivel este de a trimite pachetele sursă din orice rețea către o alta, şi să facă astfel încît acestea să ajungă la destinație indiferent de ruta şi reteaua din care au fost transmise. Protocolul care guvernează acest nivel este Internet Protocol, funcțiile îndeplinite de acesta fiind determinarea şi comutarea pachetelor (gîndiți-vă la sistemul poştal).

Nivelul rețea

Numele acestui nivel este cam general şi de multe ori crează confuzie. Este nivelul care include detalii despre tehnologile LAN/WAN, precum şi toate detaliile incluse in nivelele fizic şi legătură date din modelul OSI.

De ce trebuie să învățați despre două modele cînd unul (cel mai adesea TCP/IP) ar fi suficient? Specialiştii preferă modelul OSI pentru analize mai atente şi ca fundament în orice discuție legată de rețele. Este adevarat ca TCP/IP este mai folositor pentru că este implementat în lumea reala. Ca utilizatori finali aveți de-a face numai cu nivelul Aplicație, dar cunoaşterea detaliată a nivelurilor este vitală pentru realizarea unei rețele. Este adevărat că majoritatea utilizatorilor nu ştiu mai nimic despre protocoale de rutare sau alte detalii, dar este de asemenea adevărat că aceşti utilizatori nu trebuie să realizeze rețele scalabile şi sigure (şi nici nu au de dat examene la astfel de discipline). Prin urmare, discuția legată de TCP/IP continuă în capitolele viitoare. Pînă atunci continuăm cu....

Capitolul 2: INFRASTRUCTURA REȚELEI

Clienți, servere, imprimante, baze de date relaționale, toate acestea formează componentele unei rețele locale. Acestea sînt echipamente ce realizează încapsularea şi de-capsularea datelor pentru a-şi îndeplini toate sarcinile (transmitere mail-uri, editare texte, scannare, acces la baze de date). Continuăm prezentarea tehnologiilor prin prisma modelului OSI.

2.1 Cartela de rețea

Ce relație există între o placă de rețea (**N**etwork **I**nterface **C**ard) şi un computer? NIC este o placă cu circuite ce permite comunicarea în rețea: de la şi către un computer. Denumită şi *adaptor LAN*, ea se montează într-un slot de extensie al plăcii de bază (PCI) avînd u n port prin care se realizează conectarea în rețea a computerului.

Similar altor dispozitive hardware, cartela de rețea are nevoie de un *driver* prin care să poată fi controlată. În general, orice cartelă de rețea îndeplineşte următoarele funcții:

- pregăteşte datele pentru a putea fi transmise printr-un mediu
- transmite datele
- controlează fluxul datelor dela Pc la mediul de transmisie.

Prin rețea datele circulă în serie (un bit odată) în timp ce în interiorul calculatorului circulă în paralel (16, 32 sau 64 biti odată, în funcție de bus-ul sistemului). Prin urmare, cartela de rețea trebuie să convertească datele care circulă în interiorul PC-ului în format serial.

Pentru a funcționa sub DOS sau Windows, fiecare NIC necesită o întrerupere (IRQ-Interrupt Request Line), o adresă I/O şi o adresă de memorie. Întreruperea o puteți asocia unei resurse prin care procesorul şi celelalte componente ale PC-ului îşi acordă atenție unele altora. Unele din aceste întreruperi sînt atribuite anumitor dispozitive chiar dacă acestea nu au fost încă instalate fizic în calculator (de exemplu LPT2 pentru o a doua imprimantă).

În cazul plăcilor de rețea, atribuirea unei întreruperi depinde de numărul întreruperii disponibilă pe calculator şi de numărul întreruperii prin care placa de rețea a fost proiectată să acceseze sistemul. Dacă întreruperea pe care este proiectată să lucreze placa de rețea este ocupată de alt dispozitiv, trebuie să rezolvați conflictul care apare reconfigurînd cartela să lucreze pe altă întrerupere. Detalii despre toate întreruperile unui PC găsiți la adresa http://www.pcguide.com/ref/mbsys/res/irq

Adresa de memorie (Memory I/O Address) va conține informații despre zona de memorie pe care respectivul dispozitiv şi sistemul de operare o vor folosi pentru a-şi transmite date. Intervalul uzual de adrese pe care o placa de rețea îl foloseşte este 0x240-0x360. O parte din aceste adrese sînt deja atribuite unor dipozitive. De exemplu adresa 0x278 este folosită de cel de al doilea port paralel iar 0x378 de primul. Cartele de sunet pot folosi 0x220 iar drive-urile CDROM pot folosi 0x300.

2.2 Medii de transmisie

Dacă PC-ul este dotat cu o NIC nu înseamnă că avem musai şi o reţea. Ca şi în cazul telefonului, mai este nevoie de un lement prin care PC-ul nostru să poată fi legat la reţea. În această categorie intră mediile de transmisie sau cablurile, în limbaj reţelestic. Vom face în continuare o prezentare a principalelor medii de tranmisie în funcţie de gradul lor de utilizare în practică.

Unshielded Twisted-Pair (UTP)

Acest mediu de transmisie este format din patru perechi de fire, izolate între ele. Prin torsadarea perechilor de fire apare efectul de anulare, efect ce limitează degradarea semnalelor datorită interferenţelor magnetice sau radio.

UTP-ul este un cablu uşor de instalat (are un diamtru de 0.17") şi mult mai ieftin decît alte tipuri de cabluri. Deşi este considerat cel mai rapid mediu de transmisie bazat pe cupru, este mai vulnerabil în faţa zgomotelor electrice în comparaţie cu alte categorii de cabluri.

Există numeroase tipuri de cabluri torsadate, două dintre acestea fiind mai importante pentru reţele. Cablurile UTP din categoria 3 sînt formate din două fire izolate împletite împreună. O variantă mai performantă de astfel de cabluri este categoria 5. Sînt similare celor din categoria 3 dar au mai multe răsuciri pe centimetru şi ar trebui să fie izolate cu teflon, rezultînd de aici o interferenţă redusă şi o mai bună calitate a semnalului pe distanţe mari.

Conectorul standard folosit în cazul acestui cablu este RJ-45 (*Registered Jack*), asemnător cu cel de la firul telefonic. Conectorul este construit în baya unui standard din industria telefonică, standard care precizează care fir trebuie să fie conectat pe un anumit pin al conectorului (vom reveni cu detalii).

Tip cablu	**Utilizare**
Categoria 1	Telefonie
Categoria 2	Transfer date pînă la 4Mbps
Categoria 3	Transfer date pînă la 10 Mbps
Categoria 4	Transfer date pînă la 20 Mbps
Categoria 5	Transfer date pînă la 100 Mbps

Fibra optică

Fibra optică este mediul care asigură transmiterea luminii, modulată la o anumită frecvenţă. Comparativ cu alte medii de transmisie, fibra optică este cea mai costisitoare, dar

nu este susceptibilă la interferenţe electromagnetice şi în plus asigură rate de transfer mult mai redicate decît celelalte categorii de medii.

Cablul fibră optică constă în două fibre de sticlă îmbrăcate separat într-un înveliş de plastic (materialul se numeşte Kevlar). Cele două fibre formează inima acestui mediu de transmisie, sticla din care sînt realizate avînd un grad ridicat de refracţie. Vom reveni cu mai multe detalii

Cablul coaxial

Cablul coaxial (coax) constă dintr-un înveliş protector care îmbracă două elemente conductoare: un fir de cupru îmbrăcat într-un material izolator şi o folie metalică (sau o plasă) ce acţionează ca al doilea fir din circuit. Acest al doilea element este folosit pentru a reduce interferenţele externe. Este cablul cu cea mai bună ecranare

Pentru că nu prea se mai foloseşte (cu excepţia reţelelor "foarte bătrîne") nu vom mai insista cu alte detalii asupra acestui tip de cablu.

Conectorul folosit de acest tip de cablu se numeşte BNC (Bayone-Neill-Concelman)

Shielded Twisted-Pair (STP)

Cablul shielded twisted-pair (STP) combină trei tehnici legate de transmisia datelor: *shielding* (protejarea), *cancellation* (anularea) şi *torsadarea* firelor.

Cablul STP de 100 ohm folosit în reţelele Ethernet, oferă rezistenţă atît la interferenţele electromagnetice cît şi la cele radio fără a fi un cablul prea gros.

În reţelele Token Ring se foloseşte cablul STP de 150 ohm, în care fiecare pereche de fire torsadate este izolată cu un înveliş protector pentru a se reduce posibilitatea transferului semnalului în alte fire (cross-talk). Învelişul protector folosit în cablul de 150 ohm nu face parte din circuit aşa cum se întîmplă în cazul cablului coaxial. Chiar dacă este mai scump decît UTP (Unshielded Twisted-Pair), cablul STP oferă protecţie împotriva tuturor tipurilor de interferenţe. Spre deosebire de cablul coaxial, învelişul protector nu face parte din circuitul electric. O conectare incorectă face ca învelişul protector să acţioneze ca o antenă, absorbind semnalele electrice din cablulrile aflate în vecinătate

2.3 Echipamente de transmisie a datelor

Chiar dacă acest capitol tratează subiecte legate de nivelul 1 OSI, vom extinde puţin discuţia prezentînd şi echipamentele care corespund nivelurilor 2 şi 3

Repetorul

Chiar dacă acest echipament nu are un simbol standardizat în cadrul proiectelor de reţea, îl vom folosi pe cel alăturat. Motivul? CISCO foloseşte acelaşi simbol în manualele de instruire şi în proiectele de reţele.

Termenul de repetor vine tocmai de la începuturile comunicării vizuale cînd, o persoană aflată pe un deal, repeta semnalul pe care tocmai îl primise de la o persoană aflată pe un alt deal situat în vecinătatea sa, pentru a-l transmite mai departe. Telegrafia, telefonia (mai ales

cea mobilă) folosesc repetoare de semnal pentru a asigura transmiterea informațiilor la distanțe foarte mari.

Repetoarele pot fi *single port in – single port out*, *stackable* (modulare) sau *multi port* (cunoscute mai ales sub denumirea de *hub*-uri). Ele sînt clasificate ca fiind componente de nivel 1 deoarece acționează doar la nivel de biti. **Nu uitați!** Scopul unui hub este de a amplifica şi a retransmite semnale, la nivel de bit, către un număr mai mare de utilizatori: 8,16, sau 24. Procesul prin care se realizează această funcție se numeşte concentrare.

Fiecare hub are propriul său port prin care se conectează la retea şi mai multe porturi disponibile pentru calculatoare. Unele hub-uri au un port prin care pot fi legate de o consolă, ceea ce înseamnă că sînt hub-uri gestionabile/cu management. Majoritatea însă, sînt *dumb hubs* (hub-uri proaste) deoarece doar preiau un semnal din rețea şi îl repetă către fiecare port în parte.

Switch

La prima vedere un switch seamăna foarte bine cu un hub, dar după cum vedeți, simbolul său arată un flux informațional bidirecțional.

Menirea acestui dispozitiv este de a concentra conectivitatea garantînd în acelaşi timp lățimea de bandă. Switch-ul este un dispozitiv ce combină conectivitatea unui hub cu posibilitatea regularizării traficului pentru fiecare port(acțiune realizată cu ajutorul bridge-ului). Ca manieră de lucru, el comută pachetele de pe porturile transmițătoare către cele destinatare, asigurînd fiecărui port lățimea de bandă maximă a rețelei.

Această comutare a pachetelor se face pe baza adresei MAC, ceea ce face din switch un dispozitiv de nivel 2 (gîndiți-vă la fiecare port al unui switch ca la un mini-bridge).

Router-ul

Simbolul routerului descrie foarte bine cele două funcții ale sale: selecția căii de transmitere a informațiilor şi comutarea pachetelor către cea mai bună rută.

Fizic, routerele se prezintă sub o mulțime de forme, în funcție de model şi de producător. Componentele principale ale routerului sînt interfețele prin care rețeaua proprietară se conectează la alte segmente de rețea. Din acest motiv el este considerat un dispozitiv inter-rețele.

Scopul routerului este să examineze pachetele recepționate, să aleagă cea mai bună cale de transmitere a acestora şi în final să le transfere către portul corespunzător. Pentru rețelele mari, el reprezintă cel mai important dispozitiv prin care se reglează traficul rețelei. Deciziile routerului în ceea ce priveşte selectarea căii de rutare se iau pe baza informațiilor de la nivelul 3 (adresele de rețea), motiv pentru care sînt considerate echipamente de nivel 3. De asemenea, ele asigură conectivitate pentru diferitele tehnologii ale nivelului2: Ethernet, Token Ring, FDDI.

Dacă lucrurile nu sînt prea clare încă, încercați să citiți şi materialele de la următoarele adrese: http://www.whatis.com/encapsul.htm, http://www.jyu.fi/~eerwall/packet.htm, http://www.cs.mun.ca/~donald/bsc/node13.html,http://www.erg.abdn.ac.uk/users/gorry/course/intro-pages/encapsulation.html.

Capitolul 3: Nivelul legătură date

Pînă în acest moment am discutat mai mult despre ce se întîmplă la nivel fizic într-o rețea: medii de transmisie, bitii care traversează aceste medii, componente care transmit semnale electrice şi topologii. Nivelul 1 joacă un rol important în comunicația ce apare între calculatoare, dar efortul său singular nu este de ajuns. Fiecare din funcțiile nivelului 1 are propriile limitări, dar acestea sînt eliminate prin ceea ce se întîmplă la nivelul 2:

The Institute of Electrical and Electronic Engineers (IEEE)[3] este organizația profesională care a definit standardele aplicabile în domeniul rețelelor de calculatoare:

- 802.1- modul de interconectare în retea;
- 802.2- controlul legăturii logice (LLC);
- 802.3- rețele LAN cu acces multiplu şi cu detectarea purtătoarei şi a coliziunilor CSMA / CD, sau **rețelele Ethernet**[4];
- 802.4- rețele LAN cu transfer de jeton pe magistrală (Token Bus);
- 802.5- rețele LAN cu transfer de jeton în inel (Token Ring);
- 802.6- rețele metropolitane (MAN);
- 802.11- rețele fără fir;
- 802.12-rețele LAN cu prioritate la cerere.

Conform standardului Ethernet, o rețea locală este compusă din noduri şi medii de interconectare. Nodurile pot fi împărțite în două categorii:

- **Data terminal equipment (DTE)** – sînt echipamentele care funcționează ca sursă sau destinație a cadrelor transmise prin rețea. Cel mai adesea în această categorie intră PC-urile.

- **Data communication equipment (DCE)** – sînt dispozitive intermediare care recepționează şi transmit cadrele prin rețea. Se includ în această categorie hub-urile, switch-urile, router-ele, NICi-urile sau modemurile.

În timp ce modelul OSI reprezintă teoria care a stat la baza dezvoltării rețelelor, standardele IEEE au apărut în momentul în care rețelele au devenit realitate, cînd problemele practice trebuiau rezolvate. Chiar dacă modelul OSI este folosit în continuare, cînd se vorbeşte de nivelul 2 se au în vedere şi cele două noi componente apărute în timp: LLC şi MAC:

- **Media Access Control** (MAC) – realizează tranziția în jos, către mediul fizic de transmisie
- **Logical Link Control** (LLC)[5] - realizează tranziția în sus, către nivelul rețea.

Subnivelul LLC este independent de tehnologia folosită, în timp ce MAC este dependent de tehnologia folosită.

3.1 Funcțiile MAC

[3] http://standards.ieee.org
[4] Pe larg la adresa http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/ethernet.htm
[5] Definit prin IEEE 802.2

Subnivelul LLC a fost introdus de către IEEE din nevoia de a asigura independența tehnologică a unora din funcțiile nivelului legătură date. Oarecum inconsistent în serviciile pe care le oferă protocoalelor de la nivelul rețea, subnivelul LLC comunică cu tehnologiile specifice nivelurilor dinaintea sa.

LLC preia datele protocolului rețea şi le adaugă mai multe informații de control pentru a transmite pachetele IP către destinație. Pachetul IP astfel reîmpachetat este transmis subnivelului MAC unde urmează a fi încapsulat.

Subnivelul LLC răspunde de gestionarea comunicațiilor între echipamentele de pe o singură linie/legătură a rețelei. LLC este definit prin specificațiile IEEE 802.2, specificații care se referă atît la serviciile orientate conexiune cît şi la cele fără conexiune, servicii folosite de protocoalele superioare.

Subnivelul MAC se ocupă de protocoalele pe care un calculator le foloseşte pentru a accesa mediul fizic de transmisie a datelor. Adresa MAC are o lungime de 48 de biti, şi este exprimată în hexazecimal (12 cifre). Primele 6 care formează OUI (Organizational Unique Identifer), sînt administrate de către IEEE, identificînd producătorul sau vînzătorul produsului. Celelalte 6, descriu numărul interfeței (Serial Number Interface) sau o altă valoare administrată de fiecare producător sau vînzător.

Adresa MAC este "scrisă" în memoria ROM a cartelei de rețea, de unde este apoi copiată în RAM la inițializarea cartelei. Prin urmare, dacă o cartelă este înlocuită, se va schimba şi adresa fizică a calculatorului.

Cînd un dispozitiv din cadrul unei rețele Ethernet încearcă să transmită date către alt dispozitiv, va căută să deschidă un canal de comunicație cu acesta, folosind adresa MAC: datele transmise vor transporta şi adresa MAC a destinației. Pe măsură ce datele traversează mediul fizic de transmisie, NIC-ul fiecărui calculator din rețea verifică dacă adresa sa MAC corespunde adresei destinație inclusă în pachet. Dacă adresele nu sînt identice, NIC ignoră datele din pachet, date ce continuă să circule către următoare destinație. Dacă adresele sînt identice, NIC face o copie a pachetului cu date şi plasează această copie în calculator, la nivelul legătură de date. Pachetul original va continua să circule prin rețea, către alte destinații, unde se va verifica corespondența dintre adresele MAC.

Dezavanatajul major al adresării MAC constă în faptul că aceste adrese nu au o structură strict definită: vînzătorii au OUI-uri diferite. Altfel spus, adresarea MAC nu este o adresare ierarhică, după cum se va vedea că este adresarea IP. Pe măsură ce rețeaua "creşte", acest dezavantaj devine o problemă majoră.

3.2 Încadrarea (Framing)

Framing-ul sau *încadrarea* este un mecanism prin care se obțin informații complexe, operație ce nu poate fi realizată prin simpla transmisie a biților prin mediul fizic al rețelei. Care sînt calculatoarele ce doresc să comunice între ele? Cînd începe comunicarea între două calculatoare şi cînd se termină? Cînd îi vine rîndul unui calculator să comunice?

Spuneam că nivelul fizic al unui calculator se ocupă doar de biti. În timp ce încearcă să transmită aceşti biti către destinație, nivelul fizic nu garanteză că nu există şi erori. Aici intervine nivelul legătură date. Prin încadrare, biti transmişi de nivelul fizic sînt încapsulați la nivelul 2 în unități de date ale protocolului de nivel 2 (PDU de nivel 2) sau cadre (cadru-uri).

Există mai multe tipuri de cadru-uri în funcție de standardele folosite la descrierea lor. În mod generic, un cadru este împărțit în secțiuni numite cîmpuri, fiecare cîmp fiind alcătuit din bytes:

Orice calculator conectat la o rețea trebuie să dețină un mecanism prin care să poată atrage atenția celorlalte calculatoare din rețea cu privire la transmiterea unui cadru. Acest lucru este posibil prin intermediul cîmpului *start* din formatul cadru-ului.

Orice cadru conține informații cu privire la numele calculatorului sursă (sub forma adresei MAC) şi numele calculatorului destinație (tot adresa MAC). Un cadru are însă şi cîmpuri specalizate: lungimea exactă a cadrului sau tipul său pentru a specifica protocolul de nivel 3 ce face posibilă transmiterea sa prin rețea.

Datele transmise prin rețea sînt împărțite în două componente: datele propriu zise şi un set de bytes încapsulați, denumiți *padding bytes*, sau bytes de umplere. Aceşti bytes sînt adăugați cadrului pentru ca acesta să aibă o lungime minimă şi să poată respecta intervalul de timp în care este transmis.

Informațiile conținute de un cadru sînt susceptibile de a suporta erori ce pot să aibă surse diferite. Cadrele care conțin erori sînt retransmise. Acest lucru este realizat cu ajutorul cîmpului secvență/cifră de control a cadrului. Această cifră este un număr obținut pe baza datelor din cîmpul de date al cadrului. Acesta este citit de calculatorul destinație pentru a verifica dacă cadrul recepționat este corect sau este alterat de zgomotele rețelei.

Calculatorul sursă calculează o cifra de control pe care o adaugă cadrului. La destinație, se calculează o nouă cifră de control pe baza datelor conținute de cadrul recepționat, cifră care este comparată cu cea calculată de sursa mesajului. Dacă cele două cifre de control sînt identice, datele din cadru vor fi acceptate. Dacă cifrele de control nu sînt identice, sursa va fi atenționată că trebuie să retransmită datele. Pentru ca transmisia să se termine în condiții optime, sursa mesajului trebuie să atragă atenția celorlalte calculatoare asupra momentului în care cadrul se termină.

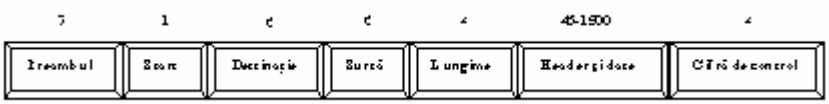
3.2.1 Standardul Ethernet

Standardul Ethernet este definit de IEEE (Institute for Electrical and Electronic Engineers) ca IEEE 802.3. Acest standard defineşte regulile pentru configurarea unei rețele Ethernet precum şi modul de interacțiune între diferitele elemente ale unei astfel de rețele (există 18 variante ale acestui standard!).

Cadrul Ethernet

Fiecare calculator echipat cu o placă de rețea Ethernet, funcționeazã independent de toate celelalte stații din retea: nu existã un control centralizat. Toate stațiile ataşate la rețea sunt conectate la acelaşi sistem de transport pentru semnal, denumit mediu de comunicație. Informația este transmisă serial, bit cu bit prin linia de comunicație către toate stațiile ataşate acesteia. Figura următoare ilustrează formatul unui cadru Ethernet aşa cum este el prezentat în specificațiile IEEE 802.3[6] (cifrele reprezintă lungimea cîmpurilor în bytes)

[6] http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/ethernet.htm#xtocid8



Conform acestui standard, cîmpurile care alcătuiesc un cadru Ethernet sînt:

- *Preambul*

Această secvență de 56 biti este folosită pentru sincronizarea transmisiei. Aceşti biti permit componentelor unei rețele să detecteze prezența unui semnal şi să înceapă citirea acestui semnal înainte de sosirea datelor conținute în cadrul respectiv. Prin intermediul acestor biti stația destinatară este avertizată cu privire la sosirea unui cadru.

- *Start cadru(SOF)*

Conform specificațiilor IEEE 802.3, byte-ul care delimitează începutul cadru-ului de restul conținutului său se termină cu doi biti consecutivi cu valoarea 1 (10101011). Aceşti biti servesc la sincronizarea recepției cadru-ului de către toate stațiile.

- *Adresă destinație şi adresă sursă*.

Primii 3 bytes ai acestui cîmp sînt precizati de către IEEE în funcție de cerințele producătorilor de echipamente pentru rețele. Următorii 3, sînt descrişi chiar de producători (parcă am mai vorbit de asta nu?). Adresa sursă este o adresă unicast (single node). Adresa destinație poate fi unicast, multicast sau broadcast.

- *Lungime/tip*

Acest cîmp indică numărul de bytes de date care urmează în cadru după acest cîmp sau tipul cadrului dacă acesta este asamblat folosind un format opțional

- *Date*

După ce procesările de la nivelurile fizic şi legătură date s-au terminat, datele conținute în cadru sînt transmise către un protocol de nivel superior care trebuie definit în cadrul acestui cîmp. Dacă datele din cadru nu ocupă cel puțin 46 bytes, vor fi inserați bytes de umplere pînă la atingerea acestei valori.

- *Cifră de control*

Acest cîmp conține o cifră de verificare pe 4 bytes, cifră ce este calculată de către dispozitivul care transmite datele, urmînd a fi recalculată de către receptor şi comparată cu originalul în scopul identificării eventualelor diferente.

Dispozitivele Ethernet trebuie să permită un interval minim de timp între două cadre care se transmit pe un mediu. Acest interval se numeşte *intercadru gap (IFG)* sau *interpacket gap (IPG)* şi foloseşte pentru pregătirea recepționării următorului cadru transmis de o stație. Acest interval este de 9,6 microsecunde pentru rețelele pe 10 Mbps, 960 nanosecunde pentru 100 Mbps şi 96 nanosecunde pentru 1 Gbps.

În continuare vom prezenta cele două protocoale prin care se controlează accesul la mediul de transmisie într-o rețea Ethernet: *half-duplex* şi *full-duplex*.

Half-Duplex Ethernet[7] (CSMA/CD Access Protocol)

[7] Comunicație semi-duplex

Half Duplexul reprezintă forma tradiţională de control în Ethernet, bazată pe protocolul CSMA/CD – Carrier Sense Multiple Access/Collision Detection (acces multiplu cu detecţia purtătoarei şi coliziunii).

Pe baza acestui protocol, staţiile care partajează acelaşi mediu şi care doresc să iniţieze o transmisie, trebuie să asculte canalul pentru a vedea dacă nu cumva transmite altcineva în acel moment. În cazul în care canalul este ocupat, staţia aşteaptă pînă la eliberarea acestuia. Atunci cînd canalul este liber, staţia transmite un cadru către toate celelalte staţii (operaţie care se numeşte *broadcast* sau *difuzare).* Există însă probabilitatea ca, imediat ce asceastă staţie începe să transmită, o altă staţie să fie pregătită de transmisie şi să asculte canalul. Dacă cadru-ul difuzat în reţea nu a ajuns încă la cea de a doua staţie, aceasta din urmă va detecta canalul ca fiind liber şi va iniţia la rîndul său o transmisie, rezultînd o coliziune.

Coliziunile pot fi detectate urmărind puterea sau lăţimea impulsului semnalului recepţionat şi comparîndu-le cu semnalul transmis. În acest caz, staţia care a iniţiat transmisia lansează în reţea o secvenţă *jam* (de blocare) de 32 biti prin care se asigură că toate celelalte staţii din reţea au fost informate cu privire la eşuarea transmisiei. Apoi staţi asursă îşi abandonează transmisia, aşteaptă o perioadă de timp şi încearcă iar dacă nici o altă staţie nu a început să transmită între timp. Acest proces se repetă pînă cînd cadrul este transmis cu succes la destinaţie.

În sinteză, principalele etape în transmiterea unui cadru sînt:

1. staţia care doreşte să transmită ascultă reţeaua cu scopul detectării prezenţei unei staţii care transmite (carrier sense – detecţia purtătoarei)
2. dacă este detectată o purtătoare activă, transmisia este amînată. Staţia continuă să monitorizeze reţeaua pînă în momentul dispariţiei purtătoarei
3. dacă nu este detectată o purtătoare activă, staţia sursă iniţiază transmiterea cadrelor
4. odată cu transmiterea cadrului, staţia sursă supraveghează mediul în vederea detectării coliziunilor.
5. dacă este detectată o coliziune, staţia sursă opreşte transmisia cadrelor şi lansează o secvenţă de blocare pentru a se asigura că toate celelalte staţii iau cunoştinţă de existenţa coliziunii.
6. după ce a transmis secvenţa de blocare, staţia sursă aşteaptă o perioadă de timp înainte de a recîncepe transmisia (de la punctul 1). Acest proces se numeşte *backoff algorithm*[8] (algoritm de regresie): se reduce probabilitatea de apariţie a coliziunilor prin adaptarea dinamică a numărului staţiilor care încearcă să transmită (interval de întîrziere generat aleatoriu)
7. dacă totuşi reapar coliziuni, intervalul de generare aleatorie creşte exponenţial. Algoritmul asigură o întîrziere minimă cînd se ciocnesc numai cîteva staţii, dar garantează că ciocnirea este rezolvată într-un interval rezonabil cînd este vorba de mai multe staţii.
8. procesul se repetă pînă cînd o staţie transmite un cadru fără coliziuni.

Un parametru important al operării în modul half duplex este *slot time* (mărimea cuantei). Acest parametru a fost definit ca avînd 512 intervale de bit (51,2 microsecunde) pentru reţelele Ethernet care operează la viteze de 10 şi 100 Mbps, respectiv 4096 intervale de bit pentru reţelele Gigabit. Mărimea cuantei se referă la intervalul de timp pe care un dispozitiv îl aşteaptă înainte de a retransmite după apariţia unei coliziuni.

Pe măsură ce traversează reţeaua, semnalele transmise suferă întîrzieri. Aceste întîrzieri reprezintă timpul necesar unui semnal să tranziteze prin componentele electronice ale reţelei. Cu cît lungimea segmentelor şi numărul de repetoare(huburi) se apropie de maximul

[8] binary exponential backoff algorithm (algoritm de regresie binară exponenţială)

admis de standardele Ethernet (2500 metri şi 4 repetoare) cu atît se măreşte şi intervalul de timp necesar unui semnal pentru a traversa reţeaua de la un capăt la altul. Acest interval de timp se numeşte *întîrziere la propagare.*

Suma dintre întîrzierea dus-întors la propagare (maximă) şi timpul necesar pentru a transmite o secvenţă de blocare sînt componentele care definesc mărimea cuantei în Ethernet.

O cuantă cu mărimea de 512 intervale de bit stabileşte mărimea minimă a unui cadru Ethernet la 64 bytes (în cazul Gigabyte, cele 4096 intervale de bit impun adăugarea unui cîmp de extensie la cadru pentru a se atinge mărimea minimă de 512 bytes). Orice cadru a cărui dimensiune este mai mică de 64 bytes este considerat *fragment de coliziune* şi este distrus în mod automat de staţia care îl recepţionează.

Mărimea cuantei impune o limită maximă în ceeea ce priveşte dimensiunea unei reţele: lungimea segementelor de cablu şi numărul repetoarelor pe o singură cale. Dacă reţeaua este dezvoltată dincolo de aceste limite apare fenomenul de *coliziune întîrziată*. Acestea sînt coliziunile care apar prea tîrziu în timpul transmiterii unui cadru pentru a mai putea fi gestionate prin funcţia de control al accesului. Cadrele afectate vor fi distruse fiind necesară reiniţierea transmisiei.

Mărimea cuantei este cea care asigură că dacă este posibil să apară o coliziune, aceasta va fi identificată în primii 512 biti transmisi sub formă de cadre (4096 pentru Gigabyte).

Revenim acum la agoritmul de regresie pentru a explica mai în detaliu ce se întîmplă. Prin intermediul acestui algoritm staţia care a iniţiat transmisia determină intervalul de timp care trebuie să treacă după apariţia unei coliziuni, înainte ca un cadru să fie retransmis. De ce este nevoie de aşa ceva? Dacă toate staţiile ar aştepta acelşi interval de timp , atunci în mod sigur va apărea o nouă coliziune. Acest lucru este evitat prin algoritmul amintit: fiecare staţie generează aleator un număr care va determina timpul cît trebuie să aştepte înainte de a trece la identificarea purtătoarei. Acest interval de timp se numeşte *întîrziere de regresie.*

După apariţia primei coliziuni, fiecare staţie aşteaptă 0 sau 1 cuante înainte de a încerca o nouă transmisie. Dacă apare o nouă coliziune intervalul de aşteptare va fi între 0 şi 3 cuante, pentru o a treia coliziune între 0 şi 7 (2^3-1). În general, după *i* coliziuni se aşteaptă între 0 şi 2^i-1 cuante. Dacă se ajunge la un număr de 10 coliziuni, intervalul de aşteptare este îngheţat la 1023 cuante. După 16 coliziuni, funcţia MAC raportează eşecul calculatorului (**excessive collision error**) iar cadrul care trebuia transmis este distrus, aplicaţia care îl folosea fiind nevoită să iniţieze o nouă transmisie.

De ce atîtea vorbe despre acest algoritm? Pentru că erorile care apar ca urmare a coliziunilor în exces dintr-o reţea reprezintă cel mai bun indiciu că reţeaua nu mai este eficientă.

Timpul necesar transmiterii unui cadru este invers proporţional cu rata de transmisie. Pentru o reţea cu o lăţime de bandă de 100Mbps, un cadru cu o dimensiune minimă este transmis într-un timp egal cu 1/10 din mărimea cuantei. Prin urmare o coliziune care apare în timpul acestei transmisii nu va putea fi detectată de staţiile care emit semnal. Este moticul pentru care diametrul maxim al unei reţele pe 10 Mbps nu poate fi utilizat în cazul reţelelor pe 100Mbps (Fast Ethernet). Soluţia în acest caz a fost reducerea diametrului reţei.

Parametru	**10 Mbps**	**100 Mbps**	**1000 Mbps**
Mărimea minimă a unui cadruMinimum	**64 bytes**	**64 bytes**	**520 bytes**

cadru size			
Diametrul maxim al domeniului de coliziune	**100 metri UTP**	**100 metri UTP** **412 metri fibrăr**	**100 metri UTP** **316 metri fibrăr**
Dimetrul maxim al domeniului de coliziune cînd se folosesc repetoare	**2500 metri**	**205 metri**	**200 metri**
Numărul maxim de repetoare pe o rută LAN	**5**	**2**	**1**

Full-Duplex Ethernet[9]

Acest al doilea mod de operare al rețelelor Ethernet depăşeşte limitările impuse prin protocolul CSMA/CD: o stație, la un moment dat, poate fie să transmită date, fie să recepționeze. Niciodată nu se întîmplă acest lucru simultan.

În cazul full-duplex, două stații pot să schimbe simultan informații dacă există o legătură care să permită acest lucru. În acest caz, *throughput-ul* agregat al rețelei se dublează!

Operarea în modul full-duplex este restricționată de respectarea mai multor criterii. În primul rînd, mediul fizic de transmisie trebuie să suporte transmiterea şi recepționarea simultană de informații, fără a exista interferențe. Mediile a căror specificații *respectă* aceste cerințe sînt: 10-Base-T, 10Base-FL, 100Base-TX, 100Base-FX, 100Base-T2, 1000Base-CX, 1000Base-SX, 1000Base-LS, and 1000Base-T (vom reveni cu detalii). Următoarele specificații *nu* suportă modul full-duplex: 10Base5, 10Base2, 10Base-FP, 10Base-FB, and 100Base-T4.

În al doilea rînd, pentru a opera în acest mod, legăturile trebuie să fie point-to-point (punct-la-punct) sau altfel spus legătura trebuie să fie direct între două stații. Atît timp cît nu există conflicte ca în cazul mediilor partajate, nu vor apărea coliziuni şi prin urmare nici protocolul CSMA/CD nu mai este necesar. Ambele stații trebuie să suporte şi să fie configurate pentru a opera în modul full-duplex.

Deşi mai sînt destule de spus, ne oprim aici cu descrierea Ethernetului nu înainte de a mai aminti despre *agregarea legături* sau *trunking-ul* disponibil în modul full-duplex. Acest lucru înseamnă că mai multe legături fizice de tip point-to-point pot fi agregate pentru a funcționa ca o singură legătură logică.

3.5.3 Cablarea IEEE 802.3

Chiar dacă despre cabluri am mai discutat în cadrul nivelului 1, ne îndreptăm din nou atenția asupra lor, dar de astă dată prin prisma standardelor IEEE 802.3. În mod obişnuit Ethernetul foloseşete doar cîteva din standardele existente în materie de cabluri: 10Base5,

[9] duplex integral

10Base2, 10BaseT, 10 BaseF, 100BaseF. Notația anterioară înseamnă că rețeaua folosește o anunmită lățime de bandă, utilizează semnalizarea în bandă de bază şi poate suporta segmente de diferite lungimi pe diferite medii de transmisie. Vom face o prezentare sintetizată a acestor standarde.

Standard	Mediul fizic	Lățime de bandă	Lungime segment	Topologie fizică	Topologie logică
10Base2	Coaxial subțire	10Mbps	185 metri	Bus	Bus
10BaseT	UTP categoria 5	10Mbps	100 metri	Star/Extended star	Bus
10BaseFL	Fibră optică multimod	10Mbps	2000 metri	Star	Bus
100BaseTX	UPT categoria 5	100Mbps	100 metri	Star	Bus
100BaseFX	Fibră optică multimod	100Mbps	2000 metri	Star	Bus
1000BaseT	UTP categoria 5	1000Mbps	100 metri	Star	bus

3.4 FDDI

Comitetul de standardizare ANSI X3T9.5 este primul autor al standardului **Fiber Distributed Data Interface (FDDI)**. După completarea tutuor specificațiilor ANSI a transmis standardul FDDI Organizației Internaționale pentru Standardizare (ISO) care a realizat o versiune internațională a standardului FDDI, versiune perfect compatibilă cu versiunea ANSI.

Chiar dacă astăzi rețelele FDDI nu sînt atît de comune precum cele Ethernet sau Token-Ring, pe măsură ce costurile de implementare se vor reduce, ele vor deveni accesbile pe o scară mai mare.

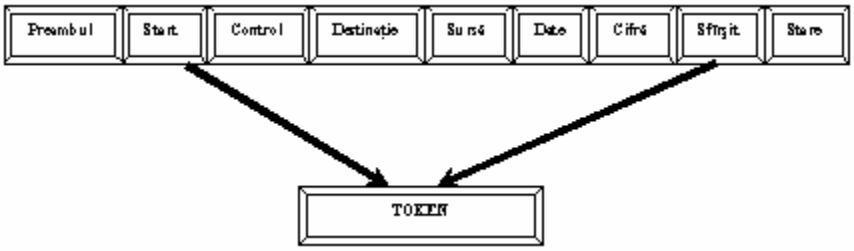
FDDI prezintă 4 specificații:

Media Access Control (MAC)- defineşte modul în care se realizează accesul la mediul fizic de transmisie, incluzînd: formatul cadru-ului, ddresarea, manipularea jetonului, algoritmul prin care se calculează CRC (verificarea redundanței ciclice) şi mecanismele pentru refacerea stării inițiale ca urmare a apariției unei erori

Physical Layer Protocol – protocolul nivelului fizic defineşte procedurile pentru codificarea/decodificarea datelor, incluzînd: cerințele ceasului (frecvența), încadrarea şi alte funcții.

Physical Layer Medium – mediul nivelului fizic defineşte caracteristiciele mediului de transmisie, incluzînd:conexiunile fibrei optice, ratele de eroare la nivel de bit, componentele optice.

Station Management- defineşte configurația stațiilor într-o rețea FDDI, incluzînd: configurația inelului, adăugarea şi eliminarea stațiilor, inițializarea, izolarea căderilor, statistici.



După cum se vede şi din poza de mai sus, cîmpurile ce alcătuiesc un cadru FDDI sînt:

- *Preambul*- pregăteşte fiecare stație pentru a putea recepționa cadru-ul
- *Delimitator de start*- indică începutul unui cadru
- *Control*- indică mărimea cîmpurilor adresă şi conține informații de control (de exemplu dacă datele sînt asincrone sau sincrone)
- *Adresă destinație*- conține o singură adresă (unicast), un grup de adrese (multicast) sau adresele tuturor stațiilor (broadcast). Adresele au 6 bytes.
- *Adresă sursă*- identifică stația care trimite cadru-ul (are 6 bytes)
- *Date*- conține informații de control sau informații destinate unui protocol de nivel superior.
- *Cifră de control*- este completată de stația sursă care calculează o CRC. Această valoare depinde de conținutul cadru-ului. Stația destinație recalculează această valoare pentru a determina dacă cadru-ul a fost modificat în timpul tranzitării prin rețea.
- *Delimitator de sfîrşit*- indică sfîrşitul cadru-ului
- *Stare*- permite stației sursă să determine apariția erorilor şi dacă la destinație cadru-ul a fost recepționat şi copiat de respectiva stație.

Strategia folosită în rețelele FDDI pentru transmiterea jetonului este similară cu cea din rețelele token-ring. FDDI permite alocarea lățimii de bandă în timp real, fapt ce le face ideale pentru o mare varietate de aplicații. Acest lucru este posibil prin cele două tipuri de trafic ce pot fi implementate: sincron şi asincron.

Traficul sincron poate consuma doar o porțiune din totalul lățimii de bandă a unei rețele (să zicem 100Mbps), în timp ce traficul asincron consumă restul. Lățimea de bandă pentru traficul sincron este alocată stațiilor care necesită transmiterea continuă a datelor (de exemplu, voce sau video). Specificațiile FDDI SMT definesc o schemă distribuită prin care se alocă lățimea de bandă.

În traficul asincron, lățimea de bandă este alocată folosind o schemă de priorități pe 8 niveluri. Fiecare stație are atribuit un nivel de prioritate asincron. Fiecare stație poate folosi la un moment dat toată lățimea de bandă asincronă. Mecanismul de prioritate poate bloca stațiile care nu folosesc lățimea de bandă sau care au un nivel de prioritate prea mic.

Specificațiile FDDI definesc două tipuri de fibră: single mod (sau mono-mod) şi multi mod. Aceste moduri se referă la fascicolul de lumină care intră în fibra optică sub un anumit unghi.

Mono-modul, după cum îi spune şi numele, permite unui singur tip de fascicol să se propage prin fibră, în timp ce multi-modul suportă mai multe tipuri de fascicole. Deoarece în multi-mod lumina care se propagă prin fibră poate parcurge distanțe diferite (în funcție de unghiul de incidență) iar semnale să ajungă la destinație la intervale diferite de timp, mono-modul oferă lățime de bandă mai mare. Acesta este şi motivul pentru care fibra

mono-mod este folosită mai ales la cablările între clădiri în timp ce fibra multi-mod se foloseşte pentru cablările intra-clădiri.

Dispozitivivele prin care se generează lumină sînt LED-urile pentru fibra multi-mod şi laserul pentru fibra mono mod.

Conform specificațiilor FDDI, pentru realizarea conexiunilor fizice se foloseşte un inel (ring) dublu. Prin fiecare din aceste inele, traficul se desfăşoară în sensuri opuse. Fizic, inelele sînt alcătuite din două sau mai multe conexiuni punct-la-punct între stațiile adiacente. Unul din cele două inele se numeşte inel principal şi este folosit pentru transmiterea datelor. Cel de al doilea inel se numeşte secundar şi este folosit în general pentru back-up.

În FDDI se întîlnesc două categoii de stații:

Single Attachement Stations (SAS) sau stații din clasa B, ataşate inelului principal prin intermediul unui concentrator care oferă conectivitate pentru mai multe astfel de stații. Concentratorul este cel care oferă continuitate rețelei în cazul întreruperilor de tensiune sau „căderilor" oricăreia dintre stații.

Dual Attachement Stations (DAS) sau stații din clasa A ataşate ambelor inele. Fiecare din aceste stații are două porturi (A şi B) prin care se conectează la ambele inele ale FDDI.

Capitolul 4: Nivelul rețea

Nivel rețea joacă un rol important în transmisia datelor: foloseşte o schemă de adresare pe care se bazează echipamentele pentru a determina care este destinația datelor pe care le transmit.

Protocoalele care nu sînt suportate de nivelul 3 pot fi folosite doar în rețelele de dimensiuni mici. Aceste protocoale folosesc de obicei un nume pentru a identifica un calculator din rețea (cum ar fi adresa MAC). Dar, pe măsură ce rețeaua se dezvoltă, organizarea acestor nume devine un calvar. Dacă vrem să interconectăm între ele două subrețele va trebui să verificăm dacă numele calculatoarelor din cele două subrețele nu sînt duplicate.

Internetul a ajuns astăzi o colecție de segmente de rețea care partajează în comun resurse informaționale. Echipamentele de nivel 3 folosite la interconectarea rețelelor sînt routerele. Acestea sînt capabile să ia decizii logice cu privire la traseul cel mai bun pe care trebuie să-l urmeze un pachet prin rețea.

4.1 Identificarea drumului optim

Drumul optim pe care îl are de parcurs un pachet este determinat în cadrul nivelului rețea. Funcția prin care este identificat drumul optim permite routerului să evalueze posibilele trasee pe care le are la dispoziție un pachet pentru a ajunge la destinație. Serviciile pentru rutare folosesc informațiile despre topologia rețelei pentru a evalua aceste rute. Determinarea drumului optim (se mai numeşte şi rutare) reprezintă procesul la care face apel un ruter pentru a alege cea mai bună cale pe care trebuie să o urmeze un pachet pentru a ajunge la destinație.

Acest proces poate fi comparat cu conducerea unei maşini de la Iaşi la Neptun, de exemplu. Avem nevoie de o hartă pentru a stabili care este traseul optim (din punct de

vedere al timpului şi combustibilului consumat) pe care trebuie să-l parcurgem pentru a ajunge la Neptun.

Adresele de reţea identifică parţial drumul folosit de un router pentru transmisia unui pachet într-o reţea: care este sursa şi destinaţia unui pachet. Gîndiţi-vă la sistemul de telefonie: fiecare oraş, ţară, continent are uncod unic de identificare. Fiecare abonat are un număr de telefon!

Dacă la nivelul 3 nu ar exista o schemă de adresare similară cu exemplul anterior, rutarea nu ar putea avea loc. Routerul are nevoie de adrese de reţea (atribuite în mod ierarhic) pentru a putea transfera în mod corespunzător pachetele către destinaţie. (Vezi şi http://www.rad.com/networks/1997/nettut/router.html)

Majoritatea dispozitivelor de reţea deţin atît o adresă MAC cît şi o adresă specifică unui anumit protocol. Cînd un calculator este mutat dintr-o reţea în alta, îşi păstrează adresa MAC dar îşi modifică adresa de reţea. Adresa MAC o putem compara cu numele noastre: odată declarat, numele unei persoane poate fi mai greu schimbat. Adresa de reţea o putem compara cu adresa domiciliului: astăzi sînteţi flotanţi. Mîine....

Principala funcţie a nivelului reţea o reprezintă găsirea celui mai bun drum pe care trebuie să-l parcurgă un pachet prin reţea, de la sursă la destinaţie. Aminteam la un moment dat că sînt două metode de adresare: ierarhizată şi non-ierarhizată.

Schema de adresare non-ierarhizată atribuie unui echipament următoarea adresă disponibilă dintr-o listă dată. Această schemă se aseamănă cu mecanismul după care se alocă CNP-urile sau mărcile angajaţilor într-o firmă.

Adresele MAC funcţionează în această manieră. Un producător de plăci de reţea primeşte o secvenţă de numere ce pot fi atribuite respectivelor dispozitive. Prima parte a fiecărei adrese resprezintă codul atribuit producătorului, iar cea de a doua parte este atribuită de vînzător, secvenţial.

Spre deosebire de aceasta, adresarea ierarhizată nu foloseşte alocarea aleatorie a unei adrese, ci mai degrabă în funcţie de localizarea unei staţii de lucru. O astfel de schemă de adresare este adresarea IP, adresare care are o structură specifică, şi care nu alocă aleator adresele.

Internet Protocol reprezintă cea mai folosită schemă de adresare ierarhică la nivelul 3. Dacă aruncăm o privire asupra modelului OSI, vom observa că pe măsură ce informaţiile străbat în jos nivelurile acestui model, datele sînt încapsulate la fiecare nivel. La nivelul reţea datele sînt transformate în datagrame, şi dacă reţeaua foloseşte adresarea IP, datele sînt transformate în datagrame IP.

4.2 Protocolul IP[10]

Elementul central al Internetului este protocolul de nivel reţea numit **IP (Internet Protocol).** Sarcina acestui protocol este de a oferi o cale pentru a transporta datagramele de la sursă la destinaţie, fără a ţine seama dacă mainile sînt sau nu în aceeaşi reţea sau dacă sînt sau nu alte reţele între ele. Teoria spune că datagramele IP pot să aibă fiecare pînă la 64Kocteţi, dar în practică acestea au în jur de 1500 octeţi. Fiecare din aceste datagrame este tranmisă prin Internet, eventual fragmentată în unităţi mai mici. Cînd toate aceste unităţi ajung la destinaţie, ele sînt reasamblate de nivelul reţea formînd datagrama originală care este pasată nivelului transport de pe maşina receptoare.

[10] Aşa cum este descris prin IETF RFC791

O datagramă IP este alcătuită dintr-o parte de *antet(header)* şi o parte de *text.* Antetul are o parte fixă de 20 octeţi şi o parte opţională cu lungime variabilă. Structura antetului IP este prezentată în continuare:

4 biti	4 biti	8 biti	16 biti	
Versiune	**IHL**	**Tip serviciu**	**Lungime totală**	
Identificare			**Flags (3biti)**	**Deplasament fragment (Fragment offset)**
Timp de viaţă (TTL)		**Protocol**	**Sumă de control**	
Adresă sursă				
Adresă destinaţie				
Opţiuni				

Cîmpul *Versiune* memorează versiunea protocolului căruia aparţine datagrama transmisă. Astfel devine posibilă tranziţia dintre versiunile aceluiaşi protocol (de la IPv4 la IPv6, de exemplu).

Cîmpul *IHL(Internet Header Length)* specifică cît de lung este antetul (spuneam că lungimea sa nu este constantă) în cuvintede 32 biti. Valoarea minimă este 5 şi se aplică atunci cînd nu sînt prezente alte opţiuni.

Tip serviciu este cîmpul care permite sursei să comunice ce tip de serviciu doreşte: fiabil, rapid sau o combinaţie. La rîndul său acest cîmp conţine un subcîmp numit *precedenţă* şi 3 flaguri (indicatori): D, T, R. Subcîmpul *precedenţă* are o lungime de 3 biti şi stabileşte priorităţile de la 0 la 7:

Valoare	Descriere
0	Normal.
1	Prioritate.
2	Imediat.
3	Flash.
4	Flash override.
5	CRITIC/ECP.
6	Internetwork control.
7	Network control

Cei trei indicatori (flaguri) permit sursei să stabilească care factori o afectează cel mai mult: întîrzierea, throughput-ul[11] (productivitatea) sau fiabilitatea. Aceste cîmpuri au fost introduse pentru a sprijini deciziile pe care le au de luat ruterele. Majoritatea ruterelor ignoră însă aceşti indicatori.

Valoare D	Descriere
0	Întărziere normală.
1	Întărziere redusă.

Valoare T	Descriere
0	**Productivitate normală.**
1	**Productivitate ridicată.**

Valoare R	Descriere
0	Fiabilitate Normală.
1	Fiabilitate ridicată.

[11] Cantitatea de date pe care un protocol de la nivelul transport o poate trasfera pe unitatea de timp

Cîmpul *lungime totală* se referă la întregul conţinut al datagramei: antetul şi datele. Lungimea maximă este de 65535 octeţi. La ora actuală pot fi transmise datagrame mai mari de această dimensiune doar în măsură în care destinatarul este capabil să le accepte.

Prin intermediul cîmpului *identifiare* destinatarul unei datagrame determină cărei datagrame aparţine un anumit pachet. Toate fragmentele unei datagrame conţin aceeaşi valoare de identificare.

Cîmpul *deplasamentul fragmentului* este precedat de două *indicatoare: DF* şi *MF*. DF (*Don't Fragment*) indică ruterelor să nu fragmenteze o datagramă deoarece calculatorul destinaţie nu este capabil să le asamblaze la loc. Toate calculatoarele trebuie să accepte fragmente de 576 octeţi sau mai mici.

Valoare DF	Descriere
0	Fragmentează dacă este necesar.
1	Nu fragmenta.

MF (More Fragments) este indicatorul care arată dacă toate frgamentele unei datagrame au ajuns la destinaţie. Toate fragmentele, cu excepţia ultimului au acest indicator activat.

Valoare MF	Descriere
0	Acesta este ultimul fragment.
1	Urmează alte fragmente

După aceşti indicatori spuneam că urmează *deplasamentul fragmentului* care indică locul fragmentului curent în cadrul datagramei. Toate fragmentele unei datagrame, cu excepţia ultimului, trebuie să fie multipli de 8 octeţi. Cum acest cîmp are o lungime de 13 biti înseamnă că există maxim 2^{13} fragmente pe datagramă (8192) şi o lungime maximă a acesteia de 65536 octeţi!

Timpul de viaţă este un contor folosit pentru a limita durata de viaţă a pachetelor. Acest timp este măsurat în secunde, avînd o valoare maximă de 255 secunde. Prin intermediul său se previne ca un pachet să circule la infinit prin reţea. În practică TTL contorizează doar hop-urile (salturile, ruterele) dintr-o reţea în alta.

După ce reasambleză datagramele, nivelul reţea trebuie să ştie ce să facă mai departe cu aceasta. În acest moment intervine cîmpul *protocol* care spune nivelului reţea cărui proces de transport trebuie pasată datagrama (în continuare sînt prezentate cele mai cunoscute protocoale[12]).

Valoare	Protocol
1	ICMP- Internet Control Message Protocol.
2	IGMP-Internet Group Management Protocol. RGMP- Router-port Group Management Protocol.
6	TCP- Transmission Control Protocol.
8	EGP- Exterior Gateway Protocol.
9	IGRP –Interior Gatewaz Routing Protocol.
17	UDP- User Datagram Protocol.
41	IPv6 over IPv4.

[12] Numerotarea protocoalelor este definită prin RFC1700

88	EIGRP.
89	OSPF- Open Shortest Path First Routing Protocol. MOSPF- Multicast Open Shortest Path First.
255	Reserved.

Suma de control a antetului trebuie recalculată de fiecare dată cînd antetul unei datagrame se modifică (de obicei la trecerea dintr-o rețea în alta) şi detectează erorile generate memoria ruterelor.

Despre cîmpurile *adresă sursă* şi *adresă destinație* discutăm în paragrafele următoare. Ele indică cine este la originea datagramei şi cine este destinatarul acesteia.

Cîmpul *opțiuni*, a cărei lungime este variabilă a fost inclus pentru a permite dezvoltarea versiunilor viitoare ale protocolului.

Cele mai importante opțiuni sînt:

Opțiune	**Valoare**	**Lungime**	**Descriere**
2 - Securitate	130	11	Cît de secretă este datagrama
3 – Dirijare aproximativă pe baza sursei	131	variabilă	Lista ruterelor care nu trebuie sărite
4 – Amprenta de timp	68	Variabilă	Fiecare ruter îşi adaugă adresa şi o adresă de timp
7 – Înregistrează calea	7	Variabilă	Fiecare ruter îşi adaugă adresa
9- Dirijare strictă pe baza sursei	137	Variabilă	Indică calea completă de parcurs

4.3 Adresa IP şi clasele de adrese

O adresă IP conține informațiile necesare pentru a transporta un pachet cu date prin rețea şi este reprezentată printr-un număr binar cu o valoare egală cu 32 biti. O manieră uşoară în care puteți citi o adresă IP presupune împărțirea adresei în patru octeți, fiecare octet conținînd 8 biti. Valoarea maximă a fiecărui octet (în zecimal) este 255.

Porțiunea network din cadrul unei adrese IP identifică rețeaua căreia aparține un echipament. Porțiunea host a adresei identifică în mod unic dispozitivul conectat la rețea. Deoarece o adresă IP este alcătuită din patru octeți separați prin punct, primul , al doilea sau al treilea dintre aceştia pot fi folosiți pentru a identifica rețeaua din care face parte un dispozitiv. La fel şi pentru identificarea dispozitivului în sine.

Există trei clase de adrese IP comerciale, clase gestionate de InterNIC: clasa A, B sau C (mai există D şi E dar acestea nu sînt comerciale). Clasa A este rezervată de InterNIC organizațiilor guvernamentale (mai mult guvernelor) din lumea întreagă; clasa B este rezervată organizațiilor medii-mari, iar clasa C este rezervată oricărui alt tip de organizație.

Cînd o adresă din clasa A este scrisă în format binar, primul bit este întotdeauna *0*. Primii doi biti ai unei adrese din clasa B sînt *10*, iar primii trei biti ai unei adrese din clasa C sînt întotdeauna *110*.Un exemplu de adresă IP din clasa A: 124.95.44.15.. Primul octet (124) identifică numărul rețelei atribuit de InterNIC. Administratorul acestei rețele va atribui valori pentru restul de 24 biti. O manieră uşoară prin care puteți să recunoaşteți dacă un dispozitiv face parte dintr-o rețea de clasă A, presupune să analizați primul octet al adresei IP. Numerele din primul octet al adreselor din clasa A sînt cuprinse între 0 şi 127.

Toate adresele IP din clasa A folosesc doar primii 8 biti pentru a identifica porțiunea "network" din cadrul unei adrese. Restul de trei octeți din cadrul adresei sînt rezervați porțiunii "host" din cadrul adresei. Cea mai mică adresă ce poate fi atribuită unui host va avea toți biții din cadrul ultimilor trei octeți la valoarea 0 Cel mai mare număr ce poate fi atribuit porțiunii host va avea toți bitii din ultimii trei octeți la valoarea 1.

Orice rețea care face parte dintr-o clasă A de adrese IP poate să conțină 2^{24} host-uri (adică 16.777.214). Un exemplu de adresă din clasa B: 151.10.13.28. Primii doi octeți identifică numărul rețelei atribuit de InterNIC. Administratorul unei astfel de rețele poate să atribuie valori următorilor 16 biti. Cînd vreți să recunoaşteți dacă o adresă este din clasa B analizați primii doi octeți ai adresei. Aceste adrese au întotdeauna valori cuprinse între 128-191 pentru primul octet şi între 0-255 pentru cel de al doilea octet.

Toate adresele din clasa B folosesc primii 16 biti pentru a identifica porțiunea "network" din cadrul unei adrese. Ultimii 2 octeti sînt rezervați porțiunii "host". Orice rețea care foloseşte adrese din clasa B poate atribui 2^{16}(65.534) adrese IP echipamentelor care sînt ataşate acesteia.

O adresă din clasa C: 201.110.213.28. Primii trei octeți identifică numărul rețelei atribuit de către InterNIC. Administratorul de rețea poate atribui valori doar ultimului octet. Cum puteți recunoaşte o adresă din clasa C? Analizați primii trei octeti: primul octet ia valori între 192-223, al doilea şi al treilea octet pot să ia valori între 1-255. Toate adresele din clasa C folosesc primi 24 biti pentru a identifica rețeaua din care face parte un dispozitiv. Doar ultimul octet este rezervat porțiunii "host" Orice rețea care foloseşte adrese din clasa C poate aloca 2^8(254) adrese echipamentelor ataşate acesteia.

Orice adresă IP identifică un echipament într-o rețea şi rețeaua căruia aparține. Dacă spre exdemplu calculatorul vostru vrea să comunice cu altul din rețea, ar trebui să ştiți adresa IP al celui din urmă. De fapt ar trebui să ştiți adresele tuturor calculatoarelor cu care vreți să comunicați. Ar fi complicat, nu? Din fericire acest neajuns este rezolvat de alții.

Adresele IP care au toată porțiunea de host cu valoarea 0 sînt rezervate ca *adrese de rețea*. De exemplu o adresă din clasa A 113.0.0.0 rezprezintă adresa IP pentru rețeaua 113. Un ruter va folosi această adresă pentru a transmite datele în Internet.

Să luăm ca exemplu o adresă din clasa B. Primii doi octeți nu pot fi zero pentru că valorile lor sînt atribuite de InterNIC şi reprezintă numerele rețelor respective. Doar ultimii doi octeți pot fi 0, deoarece numerele din aceşti octeți reprezintă numărul host-urilor şi sînt rezervate dispozitivelor ataşate respectivei rețele. Pentru a putea comunica cu toate dispozitivele din rețea, adresa IP trebuie să conțină 0 în ultimii doi octeți. O astfel de adresă ar fi de exemplu 176.10.0.0.

Cînd se transmit date către toate echipamentele dintr-o rețea trebuie creată o adresă de broadcast (difuzare). Broadcast-ul apare cînd stația sursă transmite date către toate celelalte dispozitive din rețea. Dar pentru a fi sigură că toate aceste dispozitive sînt "atente" la mesajul broadcast, stația sursă trebuie să folosească o adresă IP pe care să o recunoască toate celelalte echipamente din rețea. De obicei, într-o astfel de adresă, bitii din porțiunea host au toți *valoarea 1*. Pentru rețeaua folosită în exemplul anterior, adresa de broadcast va fi 176.10.255.255.(Vezi şi http://www.ralphb.net/IPSubnet)

Porțiunea "network" din cadrul unei adrese IP se numeşte identificatorul rețelei (network ID). Într-o rețea, hosturile pot comunica între ele doar dacă au acelaşi identificator de rețea. Acestea pot să partajeze acelaşi segement fizic de rețea, dar dacă au identificatori de rețea diferiți, nu pot comunica decît dacă există un alt dispozitiv care să realizeze conexiunea între sgementele logice ale rețelei (sau identificatorii acestora). (Puteți asemui aceşti identificatori de rețea cu codul poştal).

Porțiunea "host" din cadrul unei adrese IP se numeşte *identificator host* şi reprezintă zona prin intermediul căreia se identifică un dispozitiv dintr-o rețea. După cum am arătat deja, fiecare clasă de adrese IP permite un număr fix de hosturi. Dar nu trebuie să uitați că *prima* adresă din fiecare rețea este rezervată pentru a *identifica rețeaua*, iar ultima *adresă* este rezervată pentru *broadcast*.

4.4 Adresare IP în subrețele

De cele mai multe ori, în practică, pentru o mai mare flexibilitate, administratorii de rețea sînt nevoiți să împartă o rețea în două sau mai multe subrețele. Similar cu porțiunea "host" din cele trei clase de adrese, adresele pentru subrețele pot fi atribuite de către administratorul de rețea. Mai mult, ca şi în cazul general, adresele subrețelelor sînt unice.

Adresa pentru o subrețea include: numărul(identificatorul) rețelei, numărul subrețelei şi numărul hostului. Pentru a crea o subrețea, administratorul trebuie să "împrmute" biti din porțiunea de host a unei clase şi să-i folosească în cadrul cîmpului "subrețea". Numărul minim de biti ce pot fi împrumutați din zona host este 2! Dacă se doreşte a se împrumuta doar 1 bit pentru a crea o subrețea, atunci vom fi în situația de avea un singur număr pentru rețea (0) şi o adresă de broadcast (1). Numărul maxim de biti ce pot fi împrumutați din porțiunea host poate fi oricare cu condiția de a păstra cel puțin 2 biti pentru identificatorul de host.

Termenul tradițional de *prefix* pentru rețele extinse sau *subnet mask* sau *mask* se referă la identificatorul care spune dispozitivelor dintr-o rețea care parte dintr-o adresă IP reprezintă prefixul rețelei, care parte reprezintă numărul subrețelei şi care este numărul hostului. O mască de subrețea este o adresă IP şi are tot 32 de biti. Bitii din porțiunea network id şi subnet au valoarea 1 în timp ce bitii din porțiunea host au valoarea 0.

Ar fi foarte simplu dacă lucrurile s-ar opri aici. Din nefericire însă, cînd vorbim de subrețele IP vorbim şi de operații booleane:

- AND – similară înmulțirii
- OR – similară adunării
- NOT – schimbă bitul 1 în 0 sau 0 în 1.

Într-o rețea IP, cea mai mică adresă este adresa de rețea, sau identificatorul acesteia. Această afirmație este valabilă şi în cazul subrețelelor: adresa cea mai mică este adresa subrețelei. Ruterul este dispozitivul de rețea care realizează operații booleane (pentru a şti pe ce traseu trebuie să trimită informațiile), iar dintre acestea cea mai importantă este AND. Pentru a identifica o subrețea, ruterul "înmulțeşte" logic adresa IP şi subnet mask-ul, rezultatul obținut reprezentînd numărul rețelei/subrețelei:

Spuneam despre masca unei subrețele că este tot o adresă IP: are 32 de biti împărțiți în patru octeți. Într-o mască de rețea, toți biții din porțiunea network/subnetwork au valoarea 1, iar cei din porțiunea host 0.

Dacă nu s-au "împrumutat" biti pentru calcularea subrețelelor, masca de rețea pentru o rețea de clasă B va fi implicit 255.255.0.0. Dacă s-ar împrumuta 8 biti din porțiunea host, masca de subrețea a acestei clase ar deveni 255.255.255.0. Deoarece porțiunea host are doar doi octeți , numărul maxim al bitilor ce pot fi împrumutați pentru obținerea de subrețele este 14!

Să aruncăm o privire şi asupra clasei C. Porțiunea host a adreselor din această clasă are un singur octet. Prin urmare pot fi împrumutați maxim 6 biti pentru a crea subrețele şi minim 2. Subrețelele care conțin adresa de rețea şi adresa de broadcast nu pot fi folosite. Altfel spus dacă împrumutați un singur bit se vor crea două subrețele, dar nici una din acestea nu va putea fi folosită.

Valoarea oricărui octet este dată de numărul bitilor folositi. Valoarea zecimală maximă a fiecărui octet este 255, iar cel mai mare număr pe 8 biti, în binar, este 11111111. Dacă citim valorile zecimale ale acestor biti, de la stînga la dreapta vom obține: 128+64+32+16+8+4+2+1=255!

Primul lucru pe care trebuie să-l faceți atunci cînd doriți să creați subrețele: extindenți

	Rețea	Subrețea	Gazdă
Adresă IP gazdă 172.16.2.120	10101100 00010000	00000010	01111000
Subnet mask 255.255.255.0	11111111 11111111	11111111	00000000
Subnet	10101100 00010000 **172 16**	00000010 **2**	00000000 **0**

parțial porțiunea rețea din cadrul adresei, peste porțiunea host.

Să luăm ca exemplu adresa de clasă B 130.5.0.0, cu subnet mask-ul 255.255.255.0. Aceasta înseamnă că au fost împrumutați 8 biti pentru subrețea, iar porțiunea rețea a adresei a fost extinsă cu 8 biti.

Să luă ca exemplu şi o adresă de clasă C (sînt cele mai folosite pe la noi): 197.15.22.31 cu subnet mask-ul 255.255.255.224. Cum ultimul octet din subnet mask are valoarea 224 (11100000 în binar), înseamnă că porțiunea network a adresei a fost extinsă cu 3 biti, ajungîndu-se la un total de 27 (24+3).

De fiecare dată cînd împrumutăm biti din porțiunea host a unei adrese este important să notăm numărul subrețelelor create. Am stabilit deja că nu putem împrumuta un singur bit pentru că nu ne foloseşte la nimic. Împrumutînd 4 biti vom crea 4 subrețele. De fiecare dată cînd vom mai împrumuta un bit din porțiunea host, numărul rețelelor create va creşte cu o putere a lui 2. Dacă împrumutăm 3 biti vom obține 8 subrețele, sau altfel spus 2^3. Dacă împrumutăm 4 biti vom obține 16 subrețele, sau 2^4.

Este bine să țineți minte: *de fiecare dată cînd împrumutați cîte 1 bit din porțiunea host, numărul subrețelelor create creşte cu 2 la puterea numărului bitilor împrumutați*.

Efectul imediat al unui astfel de calcul? *De fiecare dată cînd împrumutați un bit din porțiunea host a unei adrese, numărul adreselor disponibile pentru o subrețea se reduce cu o putere a lui 2.*

Să continuăm exemplul cu o adresă din clasa C. Dacă nu folosim nici o mască de rețea înseamnă că toți bitii ultimului octet sînt folositi pentru porțiunea host. Putem astfel atribui, teoretic, 256 (2^8) adrese pentru hosturi. Să presupunem că vrem să împărțim o astfel de clasă de adrese în subrețele şi împrumutăm 1 bit din porțiunea host. Aceasta înseamnă că numărul bitilor ce pot fi alocați pentru hosturi se reduce la 7, iar numărul maxim al adreselor ce pot fi obținute se reduce la 128 (2^7).

Dacă împrumutăm 2 biti din porțiunea host, numărul bitilor ce pot fi atribuiti pentru adresele hosturilor se reduce la 6. Numărul maxim al hosturilor ce pot fi obținute pentru fiecare subrețea în parte se reduce la 64 (2^6).

Numărul adreselor ce pot fi atribuite hosturilor dintr-o subrețea este în strînsă legătură cu numarul subrețelelor create. Pentru adresele din clasa C, cu masca de rețea 255.255.255.224, înseamnă că s-au împrumutat 3 biti din porțiunea host. S-au obținut astfel 8 subrețele, fiecare cu cîte 32 hosturi (din care doar 30 utilizabile!!!).

Dacă tot am desluşit subrețelele, haideți să vedem cum se calculează numărul de rețea pentru o subrețea creată (operația se numeşte ANDing).

Vom lua ca exemplu o rețea din clasa B (într-o firmă din Iaşi): 172.16.0.0 în care vrem să împrumutăm 8 biti pentru a crea subrețele. În acest caz, masca de subrețea va fi 255.255.255.0. De la Bucureşti, cineva doreşte să transmită un mesaj către calculatorul cu adresa 172.16.2.120.

Pentru a decide unde trebuie să transmită datele, ruterul "înmulteşte" (AND) această adresă (transformată în binar) cu subnet mask-ul (tot binar). Cînd aceste două numere sînt

înmulțite, porțiunea host a adresei se pierde pentru că devine zero. Ceea ce rămîne în urma acestei operațiuni reprezintă numărul rețelei, inclusiv numărul subrețelei. Prin urmare , datele vor fi transmise subrețelei 172.16.2.0, iar ultimul ruter din cadrul rețelei va şti că pachetul trebuie transmis hostului 120 din cadrul acestei subrețele.

Să presupunem acum că pentru aceeaşi rețea, 172.16.0.0 am hotarît să împrumutăm 7 biti pentru a calcula subrețele. Masca de subrețea va fi în acest caz 255.255.254.0 (cît înseamnă în binar?) De la Bucureşti se transmite un mesaj către hostul 172.16.2.160. Ruterul va înmulți adresa hostului cu subnet mask-ul. Diferența obținută reprezintă numărul rețelei şi al subrețelelor disponibile. Dacă am împrumutat 7 biti, înseamnă că obținem 126 de subrețele (128-2). Cîte hosturi va conține fiecare subrețea?

Dacă lucrurile s-ar opri aici ar fi aproape bine. Din păcate, atunci cînd decideți să creați subrețele, trebuie să aveți în vedere şi maniera în care veți optimiza aceste subrețele şi hosturile asociate lor. De ce? Am amintit deja că nu se pot folosi prima şi ultima subrețea. De asemenea nu se pot folosi prima şi ultima adresă din cadrul fiecărei subrețele: una este adresa de broadcast a rețelei iar cealaltă este adresa de rețea.

Prin urmare, atunci cînd se realizează subrețele se pierd ceva adrese. Administratorul de rețea trebuie să fie atent la procentul adreselor care se pierd în urma unor astfel de calcule.

De exemplu, cînd împrumutăm 2 biti din porțiunea host, vom obține 4 subrețele, fiecare a cîte 64 hosturi. Însă doar două din aceste rețele vor putea fi folosite, şi doar 62 hosturi pe rețea vor fi disponibile. Ce înseamnă acest lucru? Că avem la dispoziție 124 adrese în loc de 256 cîte ar fi fost înainte de împărțirea rețelei în subrețele, ceea ce se traduce printr-o pierdere de 52%!

Numărul bitilor împrumutați	**Numărul subrețelelor create**	**Numărul hosturilor pe subrețea**	**Numărul total al hosturilor**	**Procent utilizare**
2	2	62	124	49
3	6	30	180	71
4	14	14	196	77
5	30	6	180	71
6	62	2	124	49

În fiecare clasă de adrese IP, există anumite adrese care nu sînt atribuite de InterNIC. Acestea sînt denumite adrese private sau rezervate.

Adresa 127.0.0.1 este rezervată, fiind alocată calculatorului local (local host). Se mai numeşte si adresă de *loop back (buclă.*

Acest numar va fi selectat pentru accesarea serverului Web instalat pe calculatorul propriu sau pentru a testa funcționarea cartelei de rețea.

Pentru clasa A este utilizat ID 10, adică adresele de la 10.0.0.0 la 10.255.255.255. Pentru clasa B, se folosesc ID - urile de la 172.16.0.0 pînă la 172.31.0.0. Pentru clasa C sînt disponibile adresele începînd cu 192.168.0.0 pâna la 192.168.255.0 (256 de adrese).

Calculatoarele care nu se conectează la Internet, care nu folosesc un proxy server, sau care nu folosesc network address translation (NAT), pot folosi astfel de adrese. Există o mulțime de aplicații care nu necesită conectivitate externă pentru calculatoare.

Capitolul 5: Rutarea şi protocolul de rezoluție a adresei

Spuneam că în cadrul rețelelor de calculatoare se folosesc două scheme de adresare: o schemă ce foloseşte adresele MAC (care sînt adrese de nivel 2) şi o schemă ce foloseşte adrese localizate la nivelul 3 al modelului OSI (de exemplu, adresele IP). Deoarece adresele IP sînt implementate la nivelul software-ului şi se referă la rețeaua în care poate fi localizat un anumit echipament, sînt denumite şi adrese de rețea.

Ruterul conectează două sau mai multe rețele, fiecare dintre acestea trebuind să aibă un identificator de rețea unic pentru ca routarea să aibă loc. Numărul rețelei face parte integrantă din adresa IP care este asignată fiecărui echipament ataşat rețelei.

Componenta ruterului prin care se realizează conexiunea acestuia cu rețeaua se numeşte interfață (uneori şi port). În cadrul rutăriilor IP, fiecare interfață trebuie să aibă un număr de rețea (subrețea) unic.

După ce ați stabilit schema de adresare pe care o folosiți în rețea trebuie să decideți şi maniera în care veți atribui adresle IP echipamentelor din rețea. De obicei, pentru această sarcină, se folosesc două metode: atribuire statică sau dinamică. Dar indiferent de metoda folosită trebuie să țineți cont de faptul că două interfețe nu pot să aibă aceeaşi adresă IP.

Adresarea statică implică configurarea fiecărui echipament cu propria sa adresă, în mod individual. Dacă nu se ține o evidență riguroasă a adreselor folosite pot să apară probleme în mometul în care se constată duplicarea unei adrese. În rețelele care folosesc ca sistem de operare Windows-ul (9.x, 2000 sau XP), în momentul inițializării protocolului TCP/IP, sistemul de operare transmite o cerere ARP (dacă aveți răbdare, clarificăm şi această problemă) pentru a verifica dacă adresa IP nu este duplicată. Dacă se descoperă că două stații au aceeaşi adresă IP, sistemul de operare nu va inițializa suita TCP/IP şi va genera un mesaj de eroare (de genul "stația cu adresa <adresa MAC> are aceeaşi adresa IP").

Adresarea dinamică se poate face prin mai multe metode: BOOTP, RARP sau DHCP. Despre acestea cu altă ocazie.

5.1 ICMP – protocolul mesajelor de control[13]

Pentru ca două stații să poată comunica, stația sursă are nevoie atît de adresa MAC cît şi de adresa IP a stație destinație. Dacă o stație doreşte să comunice cu o alta a cărei adresă IP este cunoscută, va căuta să-i afle doar adresa MAC. În cadrul suite TCP/IP există un protocol numit ARP (Address Resolution Protocol) prin intermediul căruia se poate detecta în mod automat adresa MAC a unei stații: ARP permite identificarea adresei MAC a unui calculator care are asociată o adresă IP.

Spuneam că în cadrul IP, unitatea de bază folosită în transferul datelor o reprezintă datagrama IP. Procesarea acestor datagrame se face la nivelul software-ului. Altfel spus, conținutul şi formatul unei datgrame nu sînt dependente de hardware.

Trebuie menționat că alte protocoale de comunicație au propriul format pentru datagrame. Datagrama IP este specifică IP.

[13] Descris în RFC792

O altă componentă majoră a IP este **ICMP(Internet Control Message Protocol)**. Acest protocol este folosit de echipamentele dintr-o rețea pentru a raporta sursei problemele care au apărut în timpul transmiterii unui mesaj. Au fost definite mai multe astfel de mesaje, cele mai importante fiind cele care urmează.

Destination unrechable (destinație inaccesibilă) este mesajul folosit atunci cînd subrețeaua sau un ruter nu pot localiza destinația unei datagrame sau cînd un pachet cu bitul DF nu poate fi livrat deoarece trebuie să tranziteze o rețea cu pachete mici.

Time exceeded (Timp depăşit) este mesajul transmis cînd un pachet este eliminat ca urmare a ajungerii contorului său la zero. De obicei prin acest mesaj se identifică blocajele din rețea.

Mesajul *Parameter problem (Problemă de parametru)* arată o valoarea nepermisă într-un cîmp din antetul datagramei.

Prin *Redirect(Redirectare)* ruterul avertizează că un pachet pare a fi dirijat pe un traseu greşit.

O caracteristică a ICMP o reprezintă echo-request/echo-reply, prin care se testează dacă un pachet poate să ajungă la destinație prin "ping-uirea" acesteia.

5.2 ARP – protocolul de rezoluție a adresei[14]

Spuneam că un pachet trebuie să conțină atît adresa MAC cît şi adresa IP a destinatarului. Dacă una din aceste adrese lipseşte, datele nu vor mai fi transferate de la nivelul 3 către celelalte niveluri ale modelului OSI. În aceset caz, cele două adrese se verifică una pe cealaltă. După ce o stație identifică adresa IP a destinatarului, poate să adauge pachetului şi adresa MAC.

Există o mulțime de variante prin care un calculatoar poate determina adresa MAC pe care trebuie să o adauge datelor în timpul încapsulării. Cel mai multe dintre acestea presupun înregistrarea tuturor adreselor MAC şi IP ale echipamentelor care sînt conectate în aceeaşi rețea. Aceste înregistrări se numesc tabele **ARP (Adress Resolution Protocol**), şi prin intermediul lor o adresă IP este mapată pe adresa MAC corespunzătoare.

Tabelele ARP reprezintă secțiuni din memoria RAM a stațiilor de lucru. Fiecare calculator din rețea are propria tabelă ARP şi cînd doreşte să transmită ceva face apel la aceasta.

Cînd stația sursă transmite către o destinație a cărei adresă IP este cunoscută, se va consulta tabela ARP pentru a se localiza adresa MAC a destinației. Dacă IP localizează o astfel de înregistrare în tabela ARP (adresă IP destinație-adresă MAC destinație), va asocia adresa IP adresei MAC identificată şi va folosi aceste adrese pentru încapsularea datelor.

Ce se întîmplă însă cînd stația sursă nu identifică în tabela ARP adresa MAC a stației destinație? În acest caz, stația sursă inițiază un proces numit cerere ARP, proces care îi permite să descopere adresa MAC a stației destinație.

Cererea ARP implică crearea unui pachet care va fi trimis tuturor stațiilor din rețea. Pentru a se asigura că toate calculatoarele vor recepționarea cererea ARP, stația sursă foloseşte o adresă MAC de broadcast: conform schemei de adresare MAC, într-o astfel de adresă toate valorile sînt F.(adresa de broadcast va avea formatul FF-FF-FF-FF-FF-FF).

Deoarece pachetele care conțin cererea ARP traversează rețeaua în modul broadcast spuneam că vor fi recepționate de către toate calculatoarele. Cînd o stație recepționează un

[14] Descris în RFC826

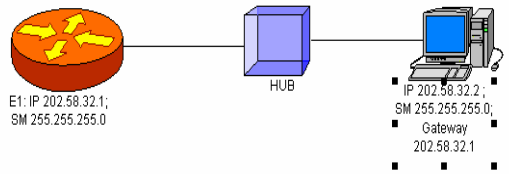
astfel de pachet îl transmite spre examinare nivelului rețea. Dacă adresa IP a stației respective corespunde adresei IP din cererea ARP, aceasta va răspunde stației sursă prin transmiterea propriei adrese MAC. Acestă operație este denumită de obicei răspuns ARP.

În momentul în care stația care a lansat cererea ARP primeşte răspuns, extrage adresa MAC din antet şi îşi actualizează tabela ARP. În acest moment, stația sursă va putea încapsula datele (la nivelul 3 şi 4) folosind ambele adrese ale stației destinație.

Ce se va întîmpla însă la destinație? Nivelul legătură date al stației destinatare extrage din antet adresa MAC pe care o compară cu adresa sa. Dacă cele două adrese sînt identice, datele sînt transferate nivelului rețea. Acesta le examinează şi "vede" că adresa IP destinație conținută în antetul IP corespunde cu adresa sa. Tot nivelul rețea extrage antetul IP şi transferă restul datelor încapsulate nivelului transport. Acest proces se repetă pînă în momentul în care restul pachetului, parțial decapsulat, ajunge la nivelul aplicație unde vor putea fi citite datele.

5.3 Default gateway şi ARP între subrețele

Am aflat deja că pentru a putea comunica, două calculatoare aflate în rețele diferite, mai au nevoie pe lîngă adresa IP şi de adresa *default gateway*: adresa IP a interfeței ruterului prin care se conectează respectivul segment de rețea. Adresa IP a gateway-ului trebuie să fie în aceeaşi rețea ca şi stația respectivă.



Dacă nu se precizează care este gateway-ul rețelei, comunicarea devine posibilă doar între calculatoarele aflate pe acelaşi segment logic de rețea. Calculatorul care doreşte să transmită date trebuie să compare adresa IP a destinatarului cu înregistrările din tabela ARP. Dacă în ARP nu se găsesc înregistrări, calculatorul sursă nu are nici o adresă IP destinație şi datele nu vor putea fi transmise.

Nu încercați să vedeți pe bucăți lucrurile prezentate pînă acum pentru că ajungem la una din problemele cele mai importante: cum comunică două calculatoare care se află în segemente de rețea diferite (atît fizic, cît şi logic). Calculatorul sursă trebuie totuşi să ştie către cine trimite datele. Şi calculatorul sursă trebuie să ştie cum să interpreteze datele primite.

Am văzut deja că ARP foloseşte pachete broadcast în anumite momente. Ruterul în schimb, nu transmite mai departe pachetele broadcast (cu excepțiile de rigoare). Un calculator care doreşte să trimită date către o stație care se află în alt segment de rețea, va trimite aceste date către gateway. Prin "înmulțirea" (AND) adresei IP cu subnet mask-ul, calculatorul sursă determină adresa de rețea a segmentului respectiv. Dacă stația destinație nu este în acelaşi segement de rețea, sursa transmite datele către gateway. Dacă stația sursă

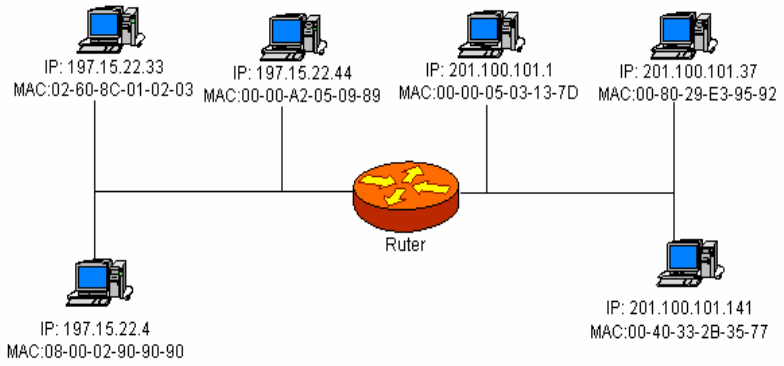
nu cunoaşte adresa MAC a gateway-ului (nu exista în tabela ARP), lansează o cerere ARP la care răspunde gateway-ul. Tabela ARP a ruter-ului conține înregistrările tuturor rețelelor conectate direct la acesta.

Mai există o variantă ARP denumită proxy ARP, folosită în rețele de dimensiuni mici şi lipsite de complexitate. În această variantă, o singură adresa IP este mapată pe mai multe adrese MAC. Un ruter pe care rulează proxy ARP, captează pachetele ARP şi răspunde cu adresa MAC corespunzătoare.

Spuneam că interfața sau portul prin care ruterul se conectează la rețea este considerată parte a rețelei şi prin urmare are propria adresă IP. Ruterul, transmite şi recepționează datele din rețea construind tabele ARP prin care mapează adresele IP la adresele MAC ale participanților. Ruterul poate fi conectat la mai multe rețele sau subrețele.

În general, echipamentele unei rețele mapează adresele IP la adresele MAC, doar în cazul dispozitivelor pe care le "văd" în mod regulat. Aceasta înseamnă că un anumit echipament conține astfel de informații numai cu privire la alte două dispozitive ale rețelei din care face parte. Despre ce se întîmplă în afara rețelei/subrețelei din care face parte, se cunosc foarte puține informații.

Tabelele ARP realizate de un ruter conțin informații despre toate rețelele/subrețelele conectate la acesta. Pe lîngă maparea adreselor IP la adresele MAC, ruterul mapează şi porturi/interfețe.



Ce se întîmplă dacă la un ruter ajunge un pachet a cărui destinație este o rețea la care respectivul ruter nu este conectat? Pe lîngă adresele IP şi MAC ale echipamentelor din rețelele la care este conectat, ruterul mai deține şi adresele IP şi MAC ale altor rutere. Acestea sînt folosite pentru a direcționa datele către destinația finală atunci cînd adresa destinație a unui pachet recepționat nu se află în tabela sa de rutare. Ruterul transmite acest pachet către un alt ruter care pare să conțină informații despre destinatar în tabela sa de rutare.

Tabela ARP este folosită doar pentru a rezolva cererile din rețelele locale. Cum se desfăşoară procesul de rutare cînd o stație cere servicii care nu pot fi satisfăcute de ruterul local, dar nu nici nu ştie adresa altui ruter?

În aceste situații, stația sursă lansează o cerere ARP. Ruterul care este conectat la aceeaşi rețea ca şi stația sursă preia cererea ARP şi răspunde stației respective. Acest răspuns conține adresa MAC a unui ruter care nu face parte din rețeaua locală. Dacă cererea

ARP pe care o lansează stația sursă nu ar "trece" de granițele rețelei locale, această stație nu ar putea obține informații despre adresele destinatarului.

Să recapitulăm puțin. Avem două stații (sursă şi destinație) aflate în rețele diferite (ceea ce înseamnă că identificatorii de rețea sînt diferiți). Stația sursă nu cunoaşte adresa MAC a stației destinație. Prin urmare sursa face apel la serviciile unui ruter pentru a putea transmite date destinatarului. Ruterul care oferă astfel de servicii este denumit deafult gateway.

Pentru a putea obține serviciile mai sus amintite, stația sursă încapsulează datele cu adresa MAC a ruterului. Adresa IP folosită în antetul datagramei IP este cea a stației destinatare şi nu cea ruterului. De ce? Pentru că stația sursă doreşte să transmită datele unei alte stații şi nu ruterului. Cînd ruterul recepționează datele, extrage şi analizează informațiile specifice nivelului 2, restul datelor fiind transferate nivelului rețea. Aici este examinată adresa IP destinație. Ruterul compară această adresa cu informațiile conținute în tabelele de rutare. Dacă găseşte o corespondență adresa IP-adresă MAC, şi destinatarul face parte din una din rețelele la care este ataşat, ruterul încapsulează datele cu noua adresă MAC şi le transmite destinației.

Dacă în schimb în tabele de rutare nu se regăsesc informați cu privire la destinatarul pachetului, ruterul caută adresa MAC a unui alt ruter şi îi transferă acestuia datele. Acest tip de rutare este cunoscut şi sub denumirea de rutare indirectă.

Cum "învață" un ruter atîtea informații? Prin două metode: *statică* şi *dinamică*. Spus în cîteva cuvinte, aceste lucruri ar suna cam aşa: dacă informațiile din tabele de rutare trebuie scrise de "cineva" se spune că este vorba de rutare statică; dacă ruterul poate învăța singur cum se va face rutarea, se spune că este o rutare dinamică.

Cînd se foloseşte un sau alta din cele două variante? Dacă administratorul de rețea doreşte să controleze cu "mînuțele" lui care vor fi căile folosite de ruter pentru a transmite pachetele la destinație sau dacă orice pachet va fi transmis pe un singur drum la destinație, se va folosi rutarea statică.

5.4 Protocoale rutabile şi protocoale pentru rutare.

Deoarece IP (Internet Protocol) este un protocol al nivelului rețea, poate fi rutat între rețele. Toate protocoalele care oferă suport nivelului rețea se numesc *protocoale rutabile* (IPX/SPX, AplleTalk).

Există însă protocoale care nu funcționează la nivelul 3. Acestea sînt denumite protocoale *non-rutabile*, şi dintre acestea cel mai cunoscut este protocolul NetBEUI. Datorită caracteristicilor sale (dimensiuni reduse, rapid, eficient), acest protocol este limitat să rulează doar pe un singur segment de rețea.

Ca un protocol să devină rutabil trebuie să ofere posibilitatea atribuirii unui număr de rețea ca şi un număr de host fiecărui calculator din rețea. Unele protocoale precum IPX/SPX necesită atribuirea doar a numărului de rețea deoarce ca identificator de host foloseşte adresa MAC a calculatorului. Alte protocoale necesită precizarea unei adrese complete dar şi a unei adrese pentru masca subrețelei (cazul IP).

Protocoalele *de rutare* (nu le confundați cu cele rutabile!) determină dumul pe care un protocol rutabil îl urmează pînă la calculatorul destinație. Astfel de protocoale sînt*: RIP (Routing Information Protocol), IGRP (Interior Gateway Routing Protocol), EIGRP(Enhanced Interior Gateway Protocol) sau OSPF(Open Shortest Path First).*

Aceste protocoale permit ruterele să realizeze o "hartă" care conține celelalte rutere din rețeaua Internet. Această facilitate permite de fapt rutarea (alegerea celui mai bun drum către destinație şi comutarea pachetelor către aceasta) prin includerea acestora în tabelele de rutare ale dispozitivului.

Ruterele folosesc astfel de protocoale pentru a schimba între ele tabelele de rutare şi alte informații despre rutare. Cel mai cunoscut protocol de rutare folosit pentru a transferul unor astfel de informații între routere este RIP (Routing Information Protocol).

RIP este un protocol de tip *interior gateway protocol(protocol pentru porți interioare)* care calculează distanța pe care trebuie să o străbată un pachet pînă la destinație, în termeni de cîte *hop*-uri (rutere sau subrețele) trebuie să traverseze respectivul pachet. Protocolul permite ruterului să-şi actualizeze tabelele de rutare la intervale regulate de timp, de obicei la fiecare 30 de secunde!

Există totuşi un dezavantaj. Ruterele care folosesc acest protocol sînt conectate la ruterele din imediata lor vecinătate. Acest lucru duce la creşterea traficului în rețea datorită schimbului continuu şi constant în timp de informații.

RIP permite însă unui ruter să determine ce cale să aleagă pentru transmiterea datelor. Acest lucru este posibil prin folosirea unui concept denumit *vector-distance*. Cînd datele traversează un router către o altă rețea, se consideră că ele traversează un hop. Într-un traseu în care numărul hop-urilor este 4, înseamnă că datele care vor urma această cale trebuie să treacă prin 4 routere pentru a ajunge la destinație. Dacă routerul identifică mai multe drumuri posibile pentru a transmite datele către destinație, va fi aleasă calea cu cel mai mic număr de hop-uri.

Deoarece numărarea hop-urilor reprezintă singura unitate de măsură folosită de protocolul RIP în determinarea celui mai bun drum pentru pachete, drumul selectat nu va fi în mod automat şi cel mai rapid. Cu toate acestea RIP rămîne unul dintre cele mai populare protocoale de rutare (poate şi din cauză că este unul dintre primele protocoale dezvoltate şi lumea s-a obişnuit cu el).

RIP are însă o limitare: numărul maxim de hop-uri prin care datele pot să treacă pentru a ajunge la destinație este 15! Dacă pînă la destinație sînt mai mult de 15 hopuri, ruterul va avertiza că destinația nu poate fi găsită.

La nivelul legătură date, datagramele IP sînt încapsulate în cadre şi sînt tratate ca date compacte (antetul IP nu este "văzut" separat de restul datelor). Ruterul recepționează cadru-ul, extrage antetul IP şi verifică adresa IP destinație. Apoi caută această adresă în tabela de rutare, încapsulează datele într-un cadru de nivel 2 şi le transmite interfeței corespunzătoare. În cazul în care în tabela de rutare adresa IP nu este găsită, ruterul renunță să mai transmită respectivul cadru mai departe.

Am dat acest exemplu pentru că ajungem să ne punem o întrebare: ce se întîmplă dacă pachetele recepționate de ruter nu folosesc ca protocol rutabil IP-ul? Ruterele sînt capabile să suporte multiple protocoale de rutare şi să întrețină tabele de rutare, în mod concurent, pentru mai multe tipuri de protocoale routabile. Despre acestea însă la momentul potrivit veți afla mai multe detalii

Majoritatea serviciilor oferite de o rețea se bazează pe un sistem de distribuție a datelor, sistem care nu presupune o conexiune directă între sursă şi destinație. Acest sistem tratează fiecare pachet în mod separat şi îl transmite către destinație pe propria sa rută. Astfel, pachetele pot să circule pe trasee diferite, dar la destinație vor fi reasamblate. Într-un astfel de sistem calculatorul gazdă nu este "cotactat" înainte ca sursa să transmită un pachet cu date (acest sistem este identic cu sistemul poştal).

La polul opus se află sistemele bazate pe conexiune: calculatorul sursă stabileşte o legătură cu destinatarul înainte de a trasnmite orice pachet (seamănă cu sistemul telefonic).

Prelucrările care apar în rețelele ccare nu necesită conexiuni sînt cunoscute sub denumirea mai ştiinţifică de *comutarea pachetelor*. În astfel de rețele, pe măsură ce un pachet porneşte de la sursă către destinație, poate fi comutat de la o rută la alta, şi este posibil chiar să nu ajungă la destinație! Dispozitivele de rețea calculează ruta fiecărui pachet în funcție de mai multe criterii, motiv pentru care această rută poate să difere de la un pachet la altul (spre exemplu, se ia în calcul lățimea de bandă disponibilă în momentul transmiterii).

Rețelele bazate pe conexiuni sînt cunoscute şi ca rețele cu *comutarea circuitelor*. Spuneam că acest lucru presupune stabilirea unei legături între cele două calculatoare. Toate pachetele care formează un mesaj vor fi transmise secvențial către destinație, urmînd acelaşi traseu/circuit.

Am făcut această scurtă descriere deoarece, ați intuit deja că, IP este un sistem care tratează pachetele în mod independent. Chiar dacă unele pachete se mai rătăcesc pe drum, IP se bazează pe protocoalele nivelului transport, protocoale care identifică pachetele pierdute şi lansează cereri de retransmisie (nivelul transport este cel care se ocupă cu ordonarea pachetelor la destinație)

Capitolul 6 Nivelul transport

Cînd am vorbit puțin despre acest nivel pe la începutul acestui material, am spus că principala sa responsabilitate se referă la asigurarea calității serviciilor pe care le oferă o rețea. Principala sarcină a nivelului 4 o reprezintă transportarea şi controlarea fluxului informațional de la sursă către destinație. Informațiile trebuie să ajungă în mod sigur "curate" la destinație.

Să presupunem că după vă angajați în domeniul rețelelor. Normal, veți intra în contact cu alți "meseriaşi" ai domeniului. Dar de cîte ori discutați probleme profesionale, se va întîmpla să nu înțelegeți tot ce vi se spune. Şi veți ruga interlocutorul să repete explicațiile sau să folosească alți termeni (doriți să fiti siguri că ați înteles) sau să vorbească puțintel mai rar (controlul fluxului informațional). Cam aşa stau lucrurile şi cu nivelul transport.

6.1 Protocoalele nivelului transport

Obiectivul nostru este să lămurim problemele legate de rețelele Ethernet bazate pe TCP/IP. La nivelul transport suita TCP/IP foloseşte două protocoale: TCP(protocol orientat pe conexiune) şi UDP(protocol fără conexiuni).

TCP Transmission Control Protocol[15]

Parte integrantă a suite TCP/IP, TCP reprezintă protocolul orientat conexiune care oferă transmisia datelor în modul full-duplex. În figura de mai jos este prezentată structura unui fragment TCP. Înainte de a prezenta semnificația elementelor din figură se cuvine să mai facem cîteva precizări.

[15] Descris în RFC793

Prin intermediul TCP, calculatoarele schimbă informații între ele sub formă de *segmente*. Un segment este format dintr-un antet de 20 de octeți urmat de zero sau mai mulți octeți de date. Programul decide singur care este mărimea acestor segmente ținînd cont de două situații. Fiecare segment nu trebuie să depășească cei 65535 octeți de informație utilă IP. Mai mult, fiecare rețea are o unitate maximă de transfer (MTU – maximum transfer unit) în care trebuie să încapă segementul TCP. Un segment care ajunge într-o rețea cu o MTU mai mică decît dimensiunea sa, va fi fragmentat de către ruterul respectivei rețele.

Fiecare nou segment format prin fragmentare are propriile antete TCP şi IP, acestă situație conducînd la creşterea gradului de încărcare al rețelei (fiecare segment primeşte un antet de 40 octeți de informație suplimentară).

32 biti								
Port sursă(16 biti)								Port destinație(16 biti)
Număr de secvență								
Număr de confirmare								
HLEN	Rezervat	URG	ACK	PSH	RST	SYN	FIN	Dimensiunea ferestrei
Sumă control								Indicator urgent
Opțiuni (cuvinte pe 32 biti)								
Date (opțional)								

Port sursă(16 biti) şi *port destinație(16 biti)* sînt cîmpurile care identifică punctul inițial şi pe cel final al conexiunii (în paragrful următor vom lămuri cum stă treaba cu porturile).

Cîmpul *număr de secvență(32biti)* este folosit pentru a asigura la destinație ordonarea corectă a informațiilor. *Numărul de confirmare(32 biti)* se referă la următorul octet care este aşteptat la destinație.

HLEN – lungimea antetului TCP indică numărul de cuvinte de 32 de biti care sînt conținute în antetul TCP deoarece cîmpul *opțiuni* este de lungime variabilă. Acest cîmp este urmat de 6 biti nefolositi (rezervati pentru dezvoltările ulterioare...dar de care nu a mai fost nevoie).

Cele 6 cîmpuri de 1 bit au următoarea semnificație:

- *URG(Urgent)* – indică deplasamentul în octeți față de numărul curent de secvență la care se află informația urgentă. Foloseşte îndeosebi la înlocuirea mesajelor de întrerupere
- *ACK (Confirmare)* – indică validitatea numărului de confirmare. Dacă acest bit are valoarea zero, este ignorat cîmpul număr de confrmare al segmentului respectiv.
- *PSH (Push - Forțare)* – acest bit indică destinatarului să livreze aplicației informația imediat ce este recepționată fără a o mai memora în buffer.
- *RST(Reset)* – desființează o conexiune care a devenit inutilizabilă
- *SYN(sincronizare)* – stabileşte conexiunea între calculatoare
- *FIN (Sfîrşit)* – termină o conexiune.

TCP-ul foloseşte pentru controlul fluxului informațional *ferestre glisante de dimensiune variabilă* (imediat lămurim şi aceste aspecte). Cîmpul *dimensiunea ferestrei* indică numărul de coteți care pot fi trimişi, începînd cu octetul confirmat.

Cîmpul *sumă de control* este calculat pentru antet şi informația propriu zisă. Inițial acest cîmp are valoarea zero iar cîmpul de date este completat cu un octet suplimentar nul, dacă lungimea sa este un număr impar.

Proiectat pentru a suporta facilități ulterioare, cîmpul *opțiune* este folosit mai ales pentru a indica lungimea maximă a unui segment TCP.

UDP - User Datagram Protocol[16]

UDP reprezintă protocolul nivelul transport care nu este orientat conexiune. Acest protocol este folosit pentru a transmite datagrame fără a fi nevoie de confirmarea recepției sau de garantarea transmiterii acestora. Retransmiterea datelor în caz de erori trebuie "ordonată" de alte protocoale.

32 biti	
Port sursă(16 biti)	**Port destinație(16 biti)**
Lungime UDP	**Sumă de control UDP**

UDP este proiectat pentru aplicațiile care nu trebuie să recompună segmentele cu date. Altfel spus, protocoalele de la nivelul aplicații sînt direct răspunzătoare de siguranța datelor transmise. Protocoalele care folosesc UDP sînt: TFTP, SNMP, DHCP, DNS.

6.2 Numărul porturilor

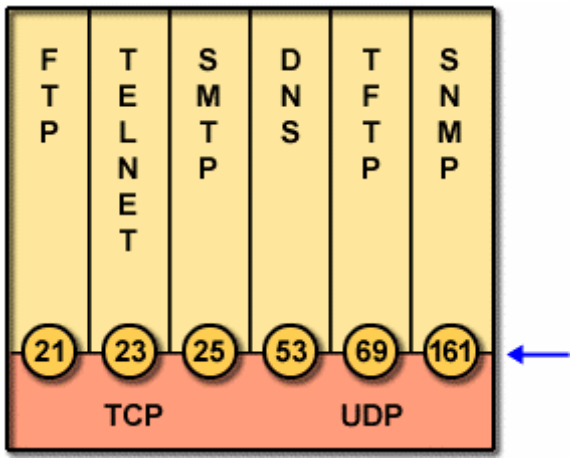
Din desenele anterioare s-a văzut că atît TCP cît şi UDP au două cîmpuri denumite numărul portului. Portul (soclu sau socket) este folosit pentru a transmite informațiile nivelurilor superioare şi pentru a urmări conversațiile care au loc în acelaşi timp în rețea.

Dezvoltatorii de aplicații sînt nevoiți să folosească numărul porturilor definite prin RFC 1700. Conversațiilor care nu folosesc aplicații cu porturi definite li se atribuie aleator numere pentru porturi. Acestea vor fi folosite ca adrese sursă şi destinație în cadrul segmentului TCP.

Pentru a discuta despre o conexiune TCP trebuie stabilită mai întîi o conexiune într un soclu de pe maşina sursă şi unul de pe maşina destinație. Un soclu poate fi folosit pentru mai multe conexiuni

Unele porturi sînt rezervate atît pentru TCP cît şi pentru UDP şi prin urmare aplicațiile care se dezvoltă nu trebuie să facă apel la acestea.

[16] Descris în RFC768

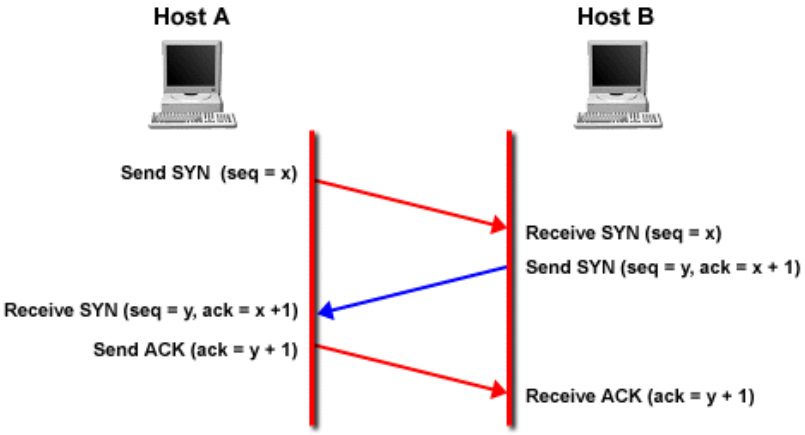


Numerele de port mai mici de 255 se numesc *porturi general* cunoscute şi sînt rezervate serviciilor standard. Porturile din intervalul 256-1023 sînt folosite de companiile publice care dezvoltă aplicații specifice iar cele peste 1024 se atribuie aleator.

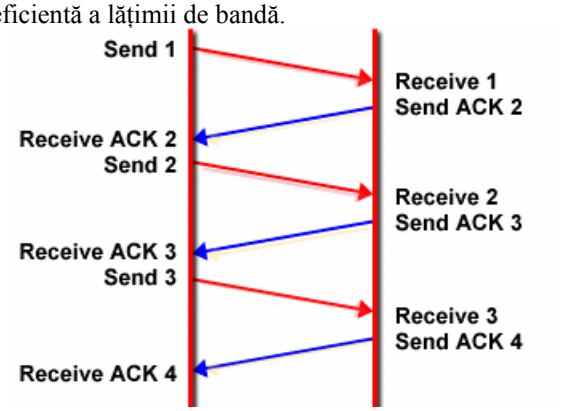
Fiecare calculator face apel la un anumit port pentru a rula o aplicație. Calculatorul sursă foloseşte un port atribuit dinamic în momentul în care lansează o cerere, port care de obiecei are un număr mai mare decît 1023. De exemplu, cînd încărcați o pagină web, la destinație se deschide portul 80 (pe care lucrează WWW-ul) în timp ce pe calculatorul vostru se deschide un port peste 1023.

6.3 Numărul secvenței şi confirmarea

Three-way handshake/open connection sau *înțelegerea în trei paşi* este folosită pentru a sincroniza conexiunea între calculatorul sursă si cel destinație înainte ca datele să fie transferate între acestea. Schimbul numerelor care reprezinta secvența este folosit pentru a se asigura că datele care se pierd datorită problemelor care pot să apară în rețea pot fi recuperate.

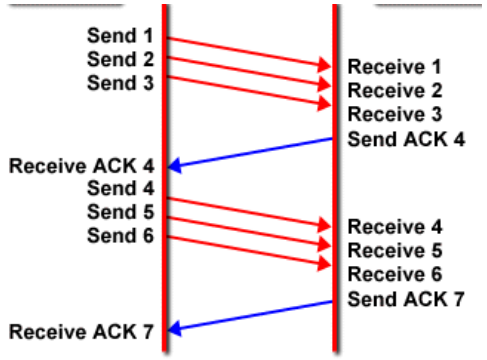


Fereastra, sau mai corect spus mărimea ferestrei, determină cantitatea de date care poate fi transmisă simultan, fără a fi nevoie de vreo confirmare. Despre ce este vorba. Cînd un calculator transmite numărul byte-ilor care determină mărimea ferestrei, trebuie să primească o confirmare din partea destinatarului înainte de a transmite alt mesaj. De exemplu, dacă mărimea ferestrei este 1, calculatorul sursă trebuie să confirme recepția fiecărui segment înainte ca sursa să-l transmită pe următorul. De aici rezultă o utilizare total ineficientă a lățimii de bandă.



Concluzia pe care o putem trage în acest moment: cu cît mărimea ferestrei este mai mare, cu atît este mai mare numărul mesajelor ce pot fi transmise înainte ca destinatarul să confirme recepția lor.

TCP folosește un sistem de confirmare în expectativă (tehnică care se numește *sliding window-fereastră glisantă*), adică numărul confirmării se referă la următorul octet care se așteaptă a fi recepționat. Partea glisantă din cadrul ferestrei se referă la faptul că mărimea ferestrei este negociată în mod dinamic în timpul unei sesiuni TCP.



Să mai facem o scurtă recapitulare. TCP ordonează segmentele într-o anumită secvență prin transmiterea către destinație a unei informări ce poartă denumirea de confirmare. Fiecare datagramă este numerotată înainte de a fi transmisă. La destinație, TCP reasamblează segmentele pentru a forma mesajul inițial. Dacă numărul unei secvențe lipsește din cadrul seriei pe care trebuia să o recepționeze stația destinație, segmentul va fi retransmis. Segmentele a căror recepție nu este confirmată într-o perioadă dată de timp trebuie retransmise.

Capitolul 7 Nivelurile sesiune şi prezentare

7.1 Cîteva consideraţii generale despre o sesiune

Nivelul sesiune este cel care coordonează aplicaţiile ce interacţionează cînd două calculatoare comunică între ele.

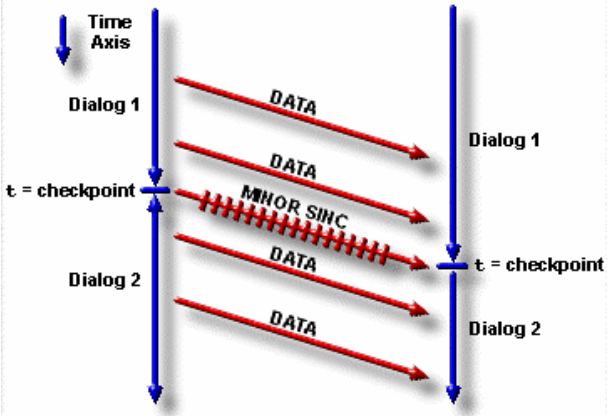
Comunicarea între două calculatoare implică derularea unor mini-conversaţii pentru a se asigura că cele două calculatoare pot efectiv comunica. În timpul acestor mini-conversaţii fiecare din participanţi joacă un rol dublu: ca şi în cazul unui client, pot să ceară la un moment dat un serviciu, dar ca şi în cazul unui server pot să ofere un serviciu. Procesul prin care se determină ce rol joacă la un moment dat unul din calculatoare se numeşte controlul dialogului.

Tot nivelul sesiune este cel care stabileşte, gestionează şi încheie sesiunile de lucru între aplicaţii. Dacă vă mai aduceţi aminte, spuneam pe la începutul acestui curs, că la nivelul sesiune trebuie să vă gîndiţi ca la dialogul uman.

Nivelul sesiune este cel care decide cînd are loc o comunicare în ambele sensuri simultan sau cînd are loc o comunicare în ambele sensuri alternativ (controlul dialogului). Dacă se permite o comunicare în ambele sensuri simultan, nivelul sesiune devine mai puţin activ în ceea ce priveşte gestionarea conversaţiei şi permite celorlate niveluri ale celor două calculatoare să controleze întregul proces. În acest caz este posibil să apară coliziuni în cadrul acestui nivel, coliziuni care sînt diferite de coliziunile care apar la nivel fizic.

Coliziunile de la nivelul sesiune se manifestă doar sub forma a două mesaje transmise unul către celălalt şi care crează confuzie fie la nivelul unui calculator, fie în ambele. Dacă aceste coliziuni nu sînt tolerate, controlul dialogului apelează la o comunicare în ambele sensuri alternativ. În acest caz se foloseşte un jeton specific nivelului sesiune, prin care cele două calculatoare stabilesc ordinea în comunicare (similar cu jetonul de la nivelul 2). În acest moment se pot pune două întrebări. Cum se realizează separarea dialogului în timpul comunicării? Cum se sincronizează comunicarea?

Separarea dialogului face apel la punctele de control (checkpoint). Aceste puncte acţionează similar cu momentul în care se declanşează AutoSave-ul Word-ului în timpul tehnoredactării. Ele separă părţile dialogului dintre cele două calculatoare. Separarea dialogului se referă la iniţierea ordinii, gestionarea şi terminarea comunicării.



În ceea ce priveşte sincronizarea comunicării, aceasta poate fi minoră sau majoră. Ce presupune sincronizarea minoră? Să aruncăm o privire la poza de mai sus.

La momentul t (momentul declanşării checkpoint-ului), nivelul sesiune al calculatorului A transmite un mesaj de sincronizare către calculatorul B. În acest moment ambele calculatoare vor executa următoarele rutine:

- realizează copii de siguranță pentru fişierele particulare
- salvează setările rețelei
- salvează setările ceasului;
- iau la cunoştință de terminarea conversației.

Sincronizarea majoră este mult mai complexă şi implică mult mai multe etape decît ar putea-o prezenta un desen (nici cel de mai sus nu are pretenția că acoperă tot ceea ce presupune sincronizarea minoră)

Şi dacă tot am pomenit în acest curs de protocoale de nivel superior, protocoalele nivelului 5 pot fi identificate în timpul login-ului sau în cadrul unei aplicații: (NFS)Network File System, (SQL)Structured Query Language, (RPC)Remote Procedure Call, X-Window System, (ASP)Apple Talk Session Protocol, DNA (Digital Network Architecture), (SCP) Session Control Protocol.

7.2 Prezentare

Nivelul prezentare este cel care răspunde de prezentarea datelor într-o formă pe care calculatorul sursă să o poată "înțelege". Acest nivel acționează ca un traducător pentru echipamentele care comunică într-o rețea şiîndeplineşte trei funcții principale:

- prezentarea datelor (sau formatul acestora);
- criptarea datelor;
- compresia datelor.

După ce primeşte datele de la nivelul aplicație, dar înainte de a le transmite nivelului sesiune, nivelul prezentare execută una sau mai multe din funcțiile prezentate anterior. La destinație, nivelul prezentare preia datele de la nivelul sesiune, execută funcțiile necesare şi apoi transferă respectivele date nivelului aplicație.

Să presupunem că stația pe care lucrați vrea să comunice cu un maincadru (!). Sistemul vostru foloseşte codurile ASCII pentru reprezentarea caracterelor, în timp ce maincadru-ul foloseşte codurile EBCDIC. Traducerea informațiilor dintr-un cod în altul este realizată cu ajutorul nivelului 6.

Dincolo de reprezentarea caracterelor, standardele nivelului 6 vizează şi modalitățile de prezentare a imaginilor grafice:

- PICT – format pentru imagini, utilizat pentru transferul imaginilor grafice QuicDraw între programele sistemelor MAC.
- TIFF – format pentru imagini bit-map cu rezoluție mare
- JPEG – formatul joint photographic experts group
- Alte cerințe ale nivelului 6 se referă la formatul de prezentare a sunetelor şi filmelor:

- MIDI –pentru sunet digital (Musical Instrument Digital Interface);
- MPEG – standard pentru compresia şi codificarea filmelor video pe suport CD etc. (Motion Picture Experts Group)
- QuickTime – standardul pentru pentru lucrul cu fişiere audio-video pe maşini MAC (diferenţă fată de QuickTime for Windows)

Ceea ce urmează este ştiut de unii dintre voi. Dar cum "repetarea este mama învăţării", voi face o scurtă prezenatre cu riscul de a vă plictisi.

Codurile ASCII şi EBCDIC sînt folosite pentru formatarea textelor. Fişierele text bazate pe coduri ASCII nu fac apel la comenzi ultra-sofisticate. Gîndiţi-vă la Notepad. E destul de simplu de scris un text cu acest utilitar. EBCDIC este un cod similar cu ASCII, singura diferenţă majoră constînd în tipul calculatorului pentru care sînt dedicate. Majoritatea fişierelor realizate cu ajutorul unui editor de texte au extensia .txt.

Un alt format destul de utilizat în cazul fişierelor este formatul binary(binar). În aceste fişiere datele sînt codificate astfel încît nu pot fi citite decît cu aplicaţii specifice. Un exemplu de program care foloseşte fişiere binare este FTP-ul.

Aminteam mai înainte de cîteva formate folosite în cazul fişierelor grafice. În Internet, cele mai folosite formate de fişiere pentru afişarea imaginilor sînt: GIF(Grafic Interchange Format) şi JPEG.

În categoria fişierelor binare intră şi formatul de fişiere multimedia: capabile să memoreze sunte şi imagini la un loc. Fişierele audio pot fi ascultate prin două metode: fie mai întîi sînt descărcate şi apoi ascultate, fie sînt ascultate în timp ce sînt descărcate (streaming audio). Windows-ul foloseşte formatul WAV pentru sunet şi AVI pentru imaginile animate. Pentru fişierele video, cele mai cunoscute formate sînt variantele MPEG.

Un alt format de fişiere, format ce acţionează ca un set de reguli pentru afişarea şi gestiunea documentelor de către browsere, este markup language. HTML (Kypertext Markup Language) este limbajul (nu de programare) folosit în Internet pentru afişarea paginilor.

Folosind un soft specializat, la nivelul 6 se poate realiza şî criptarea datelor. De ce este nevoie de criptare? Pentru a proteja informaţiile în timpul transmiterii lor prin reţea. Majoritatea tranzacţiilor financiare ce se derulează prin Internet fac apel la criptare. De cele mai multe ori, o astfel de aplicaţie foloseşte o cheie de criptare pentru a codifica datele într-o nouă formă şî o cheie de decriptare pentru a le aduce în forma iniţială.

Tot nivelul prezentare este cel care răspunde şi de compresia fişierelor, o tehnică prin care se reduce mărimea lor folosind algoritmi destul de complecşi.

Capitolul 8 Nivelul aplicaţie

Nivelul aplicaţie este cel mai apropiat de utiliztorul caluclatorului. Este nivelul responsabil cu identificarea partenerilor disponibili să comunice, sincronizează aplicaţiile, stabileşte proceduri pentru recuperarea datelor şi controleză integritatea acestora.

Este singurul nivel care nu oferă servicii celorlalte niveluri ale modelului OSI.

8.1 DNS – Domain Name System[17]

Cînd am prezentat nivelul reţea spuneam că foloseşte o schemă de adresare ierarhizată. Programele însă fac referire destul de rar la sistemele gazdă prin adresa lor IP. De cele mai multe ori se utilizează şiruri ASCII de genul: adim@yahoo.com, www.feaa.uaic.ro sau www.crap.ro. Chiar şi în această situaţie Internetul ştie numai de adresele binare despre care am discutat deja. Prin urmare a fost nevoie de un mecanism care să convertească şirurile ASCII în adrese binare. Pentru majoritatea dintre noi este mult mai uşor de memorat un nume deît o adresă binară. Din acest motiv, proiectantii Internetului au creat DNS (Sistemul Numelor de Domenii) care permite referirea calculatoarelor gazda cu ajutorul numelor

DNS este, practic, un soft care transformă numele (feaa.uaic.ro) în numere (193.226.30.15) si invers. Pentru a face o astfel de transformare, DNS are nevoie de cîteva informaţii. Aceste informatii sunt stocate pe mai multe calculatoare din Internet (servere DNS). În fond DNS este un exemplu tipic de baze de date distribuite. O bază de date distribuită poate fi văzută ca o sumă de fişiere memorate pe calculatoare diferite din Internet - localizate în spaţii geografice diferite. Softul pentru baza de date distribuită gestionează şi controlează întreaga colecţie de date ca pe o singură bază de date.

DNS este alcătuit din trei mari componente:

- - Spaţiul numelor de domeniu,
- - Servere de nume,
- - Resolvere.

Spatiul numelor de domenii reprezintă informaţia conţinută în baza de date distribuită din Internet. Putem să ne imaginăm această informaţie ca o structură arborescentă:

- arpa - este un domeniu Internet special, care transformă adresele IP în nume
- grupul generic sau al organizaţiilor - are etichete de domeniu compuse din trei caractere (com, edu, gov, mil)
- grupul geografic al ţărilor - are etichete de domeniu compuse din 2 caractere (us, ro, fr).

[17] Descris în RFC1034,1035

Internet Assigned Numbers Authority (IANA) este organismul care administrează la nivel mondial numele de domenii. La nivel conceptual, Internetul este divizat în domenii de nivel superior care, la rîndul lor sînt divizate în subdomenii etc.

Numele de domenii pot fi *relative* sau *absolute.* Numele abolute se termină cu punct (feaa.uaic.ro.) în timp ce cele relative nu. În ambele cazuri însă numele de domeniu se referă la un anumit nod din arbore şi la toate nodurile de sub el.

Componetele unui nume pot avea maximum 64 de caractere iar întreaga cale nu poate depăşi 255 caractere. Numele de domenii ar trebui să conţină doar litere, deşi în practică se întîlnesc situaţii care nu respectă această cerinţă. O astfel de situaţie poate crea probleme aplicaţiilor care analizează doar primul caracter al unei adrese pentru a identifica DNS-ul căreia aparţine.

Fiecare domeniu controleză maniera de alocare a subdomeniilor sale, motiv pentru care atunci cînd se crează un nou domeniu se cere permisiunea domeniului în care va fi inclus.

Serverele de nume sînt programe server care stochează informaţia DNS şi răspund cererilor adresate de alte programe. Un server de nume nu trebuie să ştie adresele celorlalte servere de nume din DNS. În schimb trebuie să ştie cum să contacteze serverele de nume rădăcină, care, la rîndul lor, trebuie să ştie numele şi adresele IP ale tuturor serverelor de nume de nivel doi. Arborele serverelor de nume este foarte larg şi cu foarte puţine niveluri. Deci, serverul nu trebuie sa transmita cererea la prea multe servere. Domeniile de responsabilitate poartă denumirea de *zone*. Organizaţia responsabilă pentru o anumită zonă poate divide mai departe zona, fragmentînd-o pînă cînd o singura persoană poate gestiona alocarea numelor. Această persoană este numită, de obicei, administrator DNS.

Deoarece serviciile de căutare DNS sunt operaţii critice (dacă un program nu poate obţine adresa IP căutată, nu poate realiza conexiunea dorită) Internet a impus, pentru fiecare zonă, un server DNS primar şi unul sau mai multe servere secundare. În general, serverele secundare conţin aceeaşi informaţie ca serverul primar. Ele sînt folosite pentru a crea „copii de siguranţă”, în cazul în care serverul primar se defectează sau este supraîncarcat cu cereri.

Un server de nume primar stochează informaţia DNS local, în fişiere speciale. Un server de nume secundar preia datele de la serverul primar al zonei, printr-un proces care poartă numele de *transfer zonal*. În general, un server secundar interoghează serverul primar o dată la câteva ore.

Resolverele sînt programe care extrag informaţiile din serverele de nume ca răspuns la cererile unor clienti. Un client contactează serverul de nume pentru zona din care face parte. Serverul examinează cererea pentru a determina dacă are autoritate pentru domeniul specificat. În caz afirmativ se face transformarea numelui în adresa IP şi se trimite răspunsul înapoi la client. În cazul în care serverul nu poate face transformarea direct, răspunsul depinde de tipul cererii trimise de client. Un client poate cere o transformare a numelor în doua moduri:

- *cu rezolvare recursivă* - serverul va contacta la rîndul lui un alt server de nume, de obicei de pe un nivel superior din arborele serverelor de nume. Acesta, la rîndul lui, va examina cererea, şi dacă nu poate să facă transformarea, va contacta un alt server. Şi tot aşa, pînă cînd va fi contactat un server care poate rezolva această cerere.
- *cu rezolvare iterativă* - serverul va comunica clientului ce server să contacteze mai departe. Clientul va adresa o cerere acestui server, trimis de serverul zonal, şi tot aşa mai departe pînă cînd cererea va ajunge la un server care va face transformarea. Cînd un server recepţionează o cerere cu rezolvare iterativă şi

nu poate traduce numele de domeniu, acesta va transmite clientului ce server să contacteze mai departe.

Fiecărui domeniu i se asociază anumite resurse (*resource records*), dintre care cea mai cunoscută este chiar adresa IP. Aceste resurse reprezintă de fapt datele care definesc conținutulDNS. Concret, o înregistrare de resurse se prezintă sub forma unui 5-tuplu codificat în binar (din rațiuni educative le vom prezenta în format ASCII):

Nume_domeniu Timp_de viață Tip Clasă Valoare

Nume_domeniu (Domain_name) indică domeniul căruia i se asociază înregistrarea fiind şi cheie primară a bazei de date (spuneam că e recomandabil ca numel să conțină doar litere!).

Atributul *timp_de_viață(time_to_live)* precizează cît de stabilă este înregistrarea respectivă: foarte stabilă (ia valoarea 86400 secunde) sau instabilă (60 secunde).

Atributul *tip* indică tipul înregistrării. O listă completă a acestor tipuri găsiți la adresa http://www.freesoft.org/CIE/RFC/1700/11.htm. În continuare noi vom prezenta cele mai importante tipuri de înregistrări.

Start of Authority (SOA) înregistrează parametrii specifici unei zone: numele serverului primar al domeniului, informațiile de contact ale administratorului, valabilitate etc.

Address (A) este înregistrarea care stochează adresa IP a sistemului gazdă.

Mail Exchanger(MX) sînt înregistrările care specifică hosturile dispuse să accepte poşă electronică pentru domeniul specificat.

Nameserver(NS) indică numele serverului pentru domeniul respectiv.

Canonical name (CNAME) precizează numele domeniului sau un alias pentru acesta.

Pointer(PTR). Diferența dintre PTR şi CNAME este doar una de semantică. În timp ce CNAME indică un alias (este în realitate o macrof definiție), pointerul face trecere la o altă locație din cadrul spațiului numelui de domeniu. Interpretarea sa depinde de contextul în care este utilizat.

Host information (HINFO) face o scurtă descriere a sistemului gazdă: maşina şsi sistemul de operare utilizat.

Întorcîndu-ne la structura înregistrărilor de resurse, cîmpul *clasa* are valoarea *IN* pentru toate informațiile Internet.

8.2 SNMP – Protocolul simplu pentru administrarea rețelelor[18]

SNMP (Simple Network Management Protocol) este protocolul nivelului aplicație proiectat pentru a oferi posibilitatea schimbului de informații de administrare între dispozitivele unei rețele. Parte a suitei TCP/IP, SNMP permite administratorilor de rețea să gestioneze performanțele unei rețele, să identifice şi să rezolve problemele care apar precum şi să planifice devoltările utlerioare ale rețelei.

În termeni simpli, despre SNMP se poate spune că este un protocol asimetric de tip *request-reply* care schimbă informații între o stație de administrare şi un agent. Agentul nu este altceva decît dispozitivul care se doreşte a fi administrat.

SNMP este alcătuit din trei elemente de bază:

- *Management stations* – stații de administrare

[18] Descris în RFC1157

- *Agent* – noduri administrate
- *Management Information Base (MIB)* – informații de administrare

Stațiile de administrare (NMS – Network Management Station) pot fi oricare din calculatoarele rețelei pe care se execută programele de administrare. Acestea comunică cu agenții prin intermediul SNMP trimițînd comenzi şi primind în schimb răspunsuri. Dialogul dintre aceste două componente are la bază MIB care reprezintă o colecție de informații organizate ierarhic, informații ce descriu obiectele care sînt administrate.

Protocolul SNMP permite stației de administrare să interogheze un agnet cu privire la starea obiectelor locale şi să le modifice dacă este necesar. Există situații în care dacă un agent observă că s-a produs un eveniment, raportează către toate stațiile de administare (*trap SNMP*). Stația de administare interoghează apoi agentul pentru a afla detalii despre evenimentul care a avut loc.

În cazul în care în rețea există dispozitive care nu sînt capabile să ruleze un agent SNMP se face apel la un *proxy agent* (agent intermediar) car eva prelua sarcinile agentului clasic.

Spuneam că nucleul SNMP-ului îl reprezintă baza de informații. Pentru a se asigura comunicarea între diferitele dispozitive ale unei rețele, este esențial ca obiectele să aibă o definiție standardizată, idenpendetă de producătorul respectivului echipament.

Limbajul standard folosit de SNMP pentru definirea obiectelor şi a regulilor de codificare se numeşte **ASN.1 (Abstract Syntax Notation One) – notația sintactică abstractă unu.**

8.3 Poşta electronică

Pe la începuturile sale, Internetul de astăzi a avut patru aplicații principale:

- posta electronica
- ştiri
- conectarea la distanta
- transferul de fisiere.

Poşta electronică nu este o aplicație chiar atît de simplă pe cît pare. E drept că puteți să o folosiți şi fără să stiți cele ce urmează!

Pentru a putea transmite un mesaj prin intermediul poştei electronice este nevoie de cîteva ingrediente: un calculator, o conexiune la rețea (modem, de exemplu), un program care permite utilizarea acestui serviciu de Internet, o conexiune la Internet (oferită de un provider – ISP- sau de un serviciu online) şi o banală adresa de e - mail.

Mesajul pe care îl transmiteți este preluat în rețeaua Internet de către un server şi apoi livrat calculatorului menționat în adresa de e - mail.

Cum este alcatuita o adresa de e - mail ? Adresa de poştă electronică este o adresă Internet formată din două părți, despărțite de caracterul @:

- prima parte a adresei reprezintă numele de conectare a persoanei căreia îi este destinat mesajul (ID_user)
- a doua parte reprezintă denumirea domeniului din care face parte persoana (identifică nodul destinație - adresa_nod)

Dacă aveți instalat un browser ca Netscape Navigator/Communicator sau Microsoft Internet Explorer , aveți şi aplicațiile necesare pentru e-mail. Există şi altele nu numai cele ale rivalilor aminitiți: Pine (pentru Unix), EudoraPro, America Online (AOL), HotCast, Calypso, Messenger.

Pentru a primi sau a trimite un mesaj, calculatorul trebuie însă să comunice cu un server de e - mail folosind un anumit protocol de livrare. Acest protocol se stabileşte de obicei în momentul configurării softului de e - mail.

POP - Post Office Protocol (protocol de poştă) este un protocol simplu utilizat pentru aducerea mesajelor dintr-o cutie poştală aflată la distanță. Scopul acestui protocol este de a aduce poşta electronică de la distanță şi de a o depozita pe calculatorul local al utilizatorului, pentru a fi citită mai tîrziu. Este cel mai vechi protocol, prima versiune a fost definitivata în anul 1984, ajungându-se în prezent la POP3.

IMAP - Interactive Mail Access Protocol (protocol interactiv de acces la poştă), este un protocol care a fost proiectat pentru a ajuta utilizatorii care folosesc mai multe calculatoare (un calculator la birou, un calculator acasă sau un notebook). În acest caz server - ul de e - mail păstreaza un depozit central de mesaje la care accesul poate fi realizat de pe orice calculator. În comparație cu protocolul POP3, IMAP nu copiază poşta electronică pe calculatorul personal al utilizatorului.

DMSP - Distributed Mail System Protocol (protocol distribuit pentru sistemul de poştă) este un protocol care permite utilizatorilor să aducă poşta electronică de pe serverul de e - mail pe un calculator şi apoi să se deconecteze de la server.

Cînd se alege un client de e - mail trebuie avute în vedere următoarele:

- ce standarde suportă: IMAP4 sau POP3 ?
- capabilitatea de lucru cu conturi de e - mail multiple
- posibilitatea de a aduce de pe server doar mesajele dorite, celelalte fiind eliminate prin filtre;
- posibilitatea de arhivare a mail - urilor, precum şi importul şi exportul textelor;
- ergonomia (interfața cu utilizatorul, modul de explicitare a erorilor intervenite, documentația)
- funcționalitatea: în ce masura clientul de e - mail îndeplineşte şi atinge cerințele utilizatorului, prin opțiunile puse la dispoziție;
- resurse necesare sistemului pentru fiecare aplicatie în parte pentru a rula optim si fara întreruperi;
- dacă suportă format HTML.

Toate programele de e - mail funcționează pe baza unor protocoale de comunicație, în afara celor de livrare, care asigură accesul la server - ul de poştă electronică precum şi livrarea mesajelor. Cele mai cunoscute standarde de poştă electronică (protocoalele de acces) sînt:

ISO - localizează activitățile de procesare a mesajelor electronice la nivelul 7 al modelului OSI. Acesta permite ca rețele diferite să poată comunica indiferent de deosebirile existente între sistemele de operare folosite

SMTP - Simple Mail Transfer Protocol (protocol simplu de transfer de poştă) este un protocol pentru transferul mesajelor între două calculatoare din rețea aflate la distanță. Este un protocol folosit în Internet, şi face parte din stiva de protocoale TCP / IP. Functioneaza împreuna cu programe de poştă electronică, oferind atît pentru client cît şi pentru server funcții de transmitere şi recepționare a mesajelor e – mail.

MHS - Message Handlig Service este un standard popularizat de catre firma Novell. Serverul MHS transmite mesaje între calculatoare care folosesc sisteme e - mail diferite.

MIME - Multipurpose Internet Mail Extensions (extensii de poştă cu scop multiplu) este un protocol prin intermediul caruia se pot transmite şi recepționa şi mesaje non ASCII: imagini, audio, video, etc.

8.4 File Transfer Protocol[19](Protocol pentru transferul fiţierelor)

FTP este protocolul care oferă facilităţi pentru transferul fişierelor pe sau de pe un calculator din reţea. De multe ori pentru această acţiune utilizatorul este nevoit să se autentifice pe calculatorul de pe care doreşte să încarce/descarce fişiere. Facilitatea cunoscută sub numele de *anonymous ftp* lucrează cu un cont public implementat pe calculatorul gazdă, numit *guest.*

O sesiune ftp presupune intercaţiunea a cinci componente soft:

Interfaţa utilizator	Oferă interfaţa de lucru şi gestionează interpretorul protocolului client
PI Client	Interpretorul protocolului client unterpretor este cel care transmite comenzle către interpretorul serverului aflat la distanţă şi gestionează transferul datelor către client
PI Server	Interpretorul protocolului server răspunde comenzilor lansate de client şi gestionează transferul datelor de pe server
Client DTP	Procesul de transfer al datelor către client răspunde de comunicarea cu procesul serverului şi fişierele locale.
Server DTP	Procesul de transfer al datelor de pe server răspunde de comunicarea cu procesul client şi fişierele flate la distanţă and the remote file system.

În timpul unei sesiuni ftp se realizează două conexiuni separate: una între PI-uri şi una între DTP-uri. Prima este cunoscută sub denumirea de **conexiune de control** iar ce de a doua se numeşte **conexiunea datelor.**

Cînd funcţionează, un server ftp *ascultă* portul 21 pentru a vedea dacă există cereri de conexiune. Alegerea portului pentru conexiunea datelor depinde de de comnezile transmise prin conexiunea de control. De obicei clientul transmite un mesaj de control prin care se indică numărul portului de pe maşina client pe care se accesptă o cerere de conexiune a datelor.

Folosirea celor două tipuri de conexiuni oferă avntajul selectării corespunzătoare a serviciilor: întîrziere minimă pentru conexiunea de control şi productivitate (throughput) maximă pentru conexiunea datelor.

În general, cînd se iniţiază un transfer prin ftp trebuie precizate următoarele aspecte:

1. Tipul fişierului.

Se specifică maniera în care datele conţinute de un fişier vor fi aduse într-un format transportabil prin reţea:

- fişiere ASCII – calculatorul care transmite fişierul îl converteşte din formatul local text în format ASCII.
- fişiere EBCDIC – similar cu ASCII
- fişiere binare (binary) – fişierul este transmis exact cum este memorat pe calculatorul sursă şi memorat la fel pe calculatorul destinaţie
- fişiere locale – folosite în mediile în care cel care transmite precizează numărul de biti/byte

[19] Descris în RFC 959

2. *Controlul formatului* – se referă la fişierele text care sînt transferate direct către o imprimantă:
   - No printing controls (default)
   - telnet printing controls
   - fortran printing controls
3. *Structura* – fişierele pot să-şi păstreze structura internă în timpul transmisie. De acest lucru se ocupă procesul pentru transferul datelor (DTP). Există trei posibilități:
   - Structura fişierului – fişierul este văzut ca un flux continuu de bytes, fără o structură internă
   - Structura înregistrării – fişierul este structura ca o serie de înregistrări (valabil în cazul fişierelor text)
   - Structura paginii (structură bloc) – fiecare pagină este numerotată pentru a putea fi transmisă în orice ordine
4. *Modul de transmitere.* Sînt trei posibilități:
   - Stream – fişierul este transferat într-o serie de bytes
   - Bloc – fişierul este transferat bloc cu bloc, fiecare cu un header
   - Compresat – se foloseşte o schemă de comprimare a secvențelor de bytes identici.

În timpul unui transfer prin ftp nu există nici un mecanism de negociere a transmisiei.

8.5 World Wide Web

Conceptul care a stat la baza WWW este conceptul de **hypertext** .

Prin hypertext se înțelege o colecție de documente unite între ele prin legături (*link*) ce permit parcurgerea acestora bidirectional.

HTTP est acronimul pentru **HyperText Transfer Protocol** sau protocolul ce stabileşte regulile de transfer a documentelor hypermedia. Aplicatiile care folosesc acest protocol sînt considerate entități abstracte din punctul de vedere al protocolului. Ele trebuie să poată formula cereri şi/sau recepționa răspunsuri (modelul client-server).

Unul dintre conceptele de bază este cel de **resursă** şi desemează un calculator, o bază de date, un document sau spu la modul general *un serviciu*. Resursa trebuie să poată fi referită corect şi fără echivoc. Pentru referirea unei resurse în Internet, se foloseşte termenul generic **URI -Uniform Resource Identifier.** Dacă se face referire la o locație spunem că avem de a face cu un **URL -Universal Resource Locator.** Dacă se face referire la un nume avem de-a face cu un **URN- Universal Resource Name**.

Protocolul se bazează pe paradigma cerere/răspuns. Clientul cere accesul la o resursă, aceasta fiind identificată prin URI, iar serverul răspunde printr-o linie de stare (conține, printre altele, un cod de succes sau eroare şi, în primul caz, urmează datele cerute). Cel mai simplu caz este acela cînd conexiunea client-server se realizează prin intermediul unei singure conexiuni. În general, există mai mulți intermediari de-a lungul conexiunii:

- **Proxy serverul** - primeşte cereri adresate unei resurse identificate prin URI, rescrie anumite părți ale mesajului, după care retrimite cererea către calculatorul adresat inițial. El se substituie, practic, clientului inițial, mesajul de răspuns fiinf primit tot de el. Dinspre server nu se mai "văd" clienții adevărați, ei fiind

reprezentati de serverul proxy. Serverul proxy trebuie înțeles ca un reprezentant al unui grup întreg de clienți, negociind cererile acestora adresate "restului lumii".

- **gateway** - este similar unui proxy, dar pe partea de server. Este un fel de cameră de primire pusă în fața unui server sau a unui grup de servere. Serverele de "după gateway" nu sînt vizibile, ele fiind reprezentate de gateway. Cererile sosite la gateway sînt dirijate spre serverul care poate răspunde cererii, sau celui mai liber dintre serverele ce pot răspunde. *Poarta* realizează şi o conversie de protocol, serverul nefiind obligat să "cunoască" protocolul "http"
- **tunnel** - tunelul este un intermediarmai puțin *inteligent*: transporta date pe care nu le înțelege. De obicei, la un capat al tunelului se află un server gateway, iar la capatul celălalt un proxy.

Adresarea unei resurse în Internet se face prin construcții de forma
protocol://[servciu].nume_dns[.nume_local/cale/subcale/nume_document

Serverul care răspunde cererilor privitoare la documente hypermedia se numeşte server WWW: "cunoaşte" protocolul http şi oferă serviciul www.

Partea a II a
Proiectarea rețelelor de calculatoare

Capitolul 10 Constrîngeri legate de tehnologie în proiectarea rețelelor

În cea de a doua parte vom încerca să prezentăm principalele etape pe care ar trebui să le urmeze oricine are curajul să se apuce de proiectat o rețea. Nu putem însă să discutăm în detaliu despre acest subiect fără a avea în vedere problemele care pot să apară într-o rețea şi modul cum pot fi acestea rezolvate.

10.1 Factori care afectează semnalele dintr-o rețea[20]

În cadrul sistemelor electronice, informațiile sînt reprezentate cu ajutorul unei cifre binare: 0 sau 1. Concret, într-un mediu de transmisie bazat pe cupru, cifra 0 este reprezentată prin 0 volți, în timp ce cifra 1 este reprezentată prin 5 volți. Prin urmare, bitul de care tot se vorbeşte este un puls electric de 0 sau 5 volți!

În cazul semnalelor optice, lucrurile nu diferă prea mult: bitul 0 este codat prin absența luminii sau o lumină cu intensitate scăzută, în timp ce bitul 1 este codat prin prezența luminii.

Nu de puține ori, rețelele sînt proiectate fără a exista cerințe legate de perfomanța acestora. Şi asta nu din vina proiectanților ci a clienților care dezvoltă sistemele fără a respecta o metodologie anume.

Propagarea

Funcționarea corectă a mecanismului prin care se realizează controlul accesului la mediu impune ca toate interfețele Ethernet să răspundă la semnalele recepționate într-o perioadă de timp dată.

Propagarea se referă la bitii (reprezentați prin energie electrică) ce traversează un mediu de transmisie. Viteza cu care aceştia se propagă depinde de materialul din care este realizat respectivul mediu, geometria cablajului şi structura acestuia, precum şi de frecvența pulsului electric al dispozitivelor prin care tranzitează semnalul.

[20]vezi şi http://www.cabletesting.com/CableTesting

Timpul necesar unui bit să traverseze de la un capăt la altul, şi înapoi mediul, se numeşte *Round Trip Time (RTT),* sau *timp de întoarcere.* Timpul de întoarcere maxim este strict limitat pentru a asigura că fiecare interfață de rețea poate recepționa toate semnalele din linia de comunicație într-o perioadă de timp specificată. Cu cît este mai lung un segment de rețea, cu atît mai mult timp îi ia unui semnal să îl parcurgă.

Funcționarea corectă a unei rețele Ethernet depinde de mediul de transmisie folosit în cadrul segmentelor şi de respectarea regulilor de proiectare.

RTT-ul este o primă măsurătoare ce se poate efectua cu privire la întîrzierile care pot să apară la un moment dat.

```
C:\>ping 193.226.23.4

Pinging 193.226.23.4 with 32 bytes of data:

Reply from 193.226.23.4: bytes=32 time=4ms TTL=62
Reply from 193.226.23.4: bytes=32 time=3ms TTL=62
Reply from 193.226.23.4: bytes=32 time=1ms TTL=62
Reply from 193.226.23.4: bytes=32 time=1ms TTL=62

Ping statistics for 193.226.23.4:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss)
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 4ms, Average = 2ms

C:\>_
```

Figura:

Timpul de propagare nu reprezintă o problemă ci o situație pe care trebuie să fiți siguri că o puteți stăpîni. Dacă timpul de propagare este prea mare, trebuie să reevaluați modul în care rețeaua va face față întîrzierilor care apar în transmisia datelor. Dacă în schimb acest timp este prea scurt, trebuie să ştiți cum să-i "frînați", sau să-i salvați temporar (buffering) astfel încît celelelalte echipamente din rețea să aibă acces la ei.

Într-o rețea Ethernet cu o lățime de bandă de 10 Mbps, timpul necesar transmiterii unui bit[21] este de 100 nanosecunde. În continuare vă prezentăm valorile întirzierilor pe un segment, valori ce pot fi folosite la calcularea întîrzierii totale (valori exprimate în *bit time*):

Tipul segmentului	**Lungimea maximă a segmentului (în metri)**	**Întîrzierea (RTT)**
10BASE5	500	0.0866
10BASE2	185	0.1026
10BASE-T	100	0.113
10BASE-FL	2000	0.1

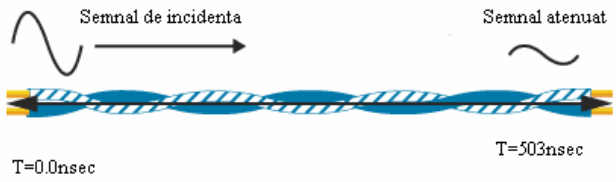
În calcularea întîrzierii totale a unei rețele se porneşte de la întirzierea cea mai mare identificată pe fiecare segement de rețea (sau cea mai mare întărziere care poate să apară pe fiecare segment). Prin însumarea respectivelor valori ale RTT se obține valoarea totală a

[21] bit time – numărul de bit transportați între dispozitivele din rețea într-o perioadă de timp dată (de obicei 1 secundă).

întirzierii din rețea. Standardele recomandă ca la această valoare să se adauge o marjă de 5 bit time. Dacă rezultatul este egal sau mai mic de 575 bit time, ruta respectivă este considerată bună. Acest lucru înseamnă că ultima stație situată pe cel mai slab segment din această rețea primeşete notificări cu privire la apariția coliziunilor într-un interval maxim de 575 bit time: 511 biti ai cadrului plus 64 din preambul şi delimitatorul de start. Dacă lucrurile merg bine cu cel mai slab segment de rețea ar trbeui să nu existe probleme cu celelalte.

Lucrurile nu se termină aici pentru că există o situație aparte care trebuie luată în calcul: ruta pentru care se realizează calculele să aibă la extremități segmente de alt tip decît restul rețelei.

Dacă valoarea întirzierii nu este specificată de producătorul cablului respectiv, trebuie calculată. Timpul de propagare este specificat de producători ca procent din viteza de propagare a luminii[22].



Întîrzierea normală la propagare pentru un cablu UTP categoria 5 este 5nanosecunde pe metru, iar cea maximă admisă este de 5,7 ns/metru. Pentru un segment cu o lungime de 100 metri întîrzierea poate fi cea din figura de mai sus

Atenuarea

Atenuarea este cuvîntul prin care se descrie pierderea de energie în timpul propagării semnalului printr-un mediu de transmisie. Aceasta înseamnă că un bit îşi pierde din voltaj, sau din amplitudine, pe măsură ce energia este transferată de la calculator către cablul prin care circulă. Pierderea se exprimă în decibeli pe kilometru, iar energia pierdută depinde de frecvența semnalului.

Dacă se aleg cu grijă mediile de transmisie, dacă se poziționează corect, atenuarea electrică poate fi redusă. Nu poate fi însă eliminată, pentru că nu poate fi eliminată rezistența materialelor. Atenuarea apare şi în cazul fibrei optice care absoarbe şi distruge o parte din pulsul de lumină, pe măsură ce acesta o traversează. Acest lucru poate fi redus prin alegerea unei lungimi de undă corespunzătoare, a culorii etc.

Rețineți însă că, deocamdată, pierderea de semnal din mediile de transmisie este inevitabilă. Ea apare şi în cazul undelor radio sau a microundelor care sînt absorbite de moleculele specifice atmosferei.

Atenuarea poate afecta rețeaua dacă nu sînt impuse unele limitări în ceea ce priveşte cablul prin care sînt transmise semnalele. Dacă un cablu este prea lung, un bit care la sursă este 1, poate să ajungă la destinație ca 0. Efectele atenuării pot fi reduse prin folosirea unor medii de transmisie de calitate superioara, structurarea rețelei şi instalarea de repetoare.

---

[22] *Nominal Velocity of Propagation*. Un cablu cu NVP de 75 transmite semnalul cu o viteză egală cu 75% din viteza luminii. Majoritatea cablurilor au NVP cuprinsă în intervalul 0,6 – 0,9c (c=viteza luminii).

Cel mai adesea, atenuarea este influențată de lungimea mediului de transmisie şi de fracvența semnalului.

Reflecția

Reflecția apare în semnalele electrice, cînd datorită discontinuității pe care bitii o întîlnesc în cablu, o parte din energia electrică se reflectă. Dacă nu este controlată în mod corespunzător, această energie poate induce confuzie printre ceilalti biti care traversează linia. În funcție de cablul şi conexiunea folosite, reflecția poate fi o problemă sau nu.

În cazul semnalelor optice, acestea se reflectă indiferent dacă sticla prezintă discontinuități sau nu. Aduceți-vă aminte că noaptea, cînd priviți pe fereastră, vă vedeți propria reflecție. Acest fenomen apare şi în cazul undelor radio şi a microundelor atunci cînd acestea circulă prin atmosferă.

Pentru ca rețeaua să ajungă la performanțe optime, este important ca mediul de transmisie să aibă o anumită impedanță care să răspundă cerințelor componentelor electrice de la nivelul NIC. În caz contrar se poate ajunge la interferențe de semnale, bitii transmişi ajungînd să fie confuji.

Zgomotul

În sistemele de comunicații, interferețele statice care distrug integritatea unui semnal ce traversează acea linie sînt referite ca zgomote. Cu alte cuvinte, este vorba de modificarea voltajului, a semnalului optic sau a undei electromagnetice ca urmare a captării de energie din alte surse. Nu există semnal electric care să nu aibă "zgomote": fiecare bit care traversează liniile de transmisie recepționează semnale de la diferite surse.

Cînd zgomotul dintr-un cablu are la origine semnalele care traversează alte fire, se spune că avem de a face cu o încrucişare (*crosstalk*). Dacă două fire aflate în apropiere unul de altul nu sînt corect ecranate, energia dintr-un fir poate fi transferată celuilat şi viceversa (gindiți-vă la principiul de funcționare al antenelor).

Zgomotele datorate liniilor de tensiune reprezintă o altă problemă crucială a rețelelor. Vrem nu vrem, în interiorul clădirilor, pereții sînt traversați de liniile de tensiune. Dacă nu sînt izolate corespunzător, acestea pot afecta traficul rețelei. Poate sînteți suprinşi dacă aflați că pînă şi "zgomotul" datorat tensiunii dintr-un monitor sau hard disk poate provoca efecte negative! Toate aceste efecte negative pot fi combătute cu ajutorul "împămîntării".

Interferențele care pot să apară între doi conductori sînt destul de greu de identificat, cu atît mai mult cu cît firele acționează de cele mai multe ori ca nişte antene pentru "zgometele" electrice (dacă nu sînt torsadate). Calculatorul trebuie să poată să discearnă între semnalele digitale şi impulsurile electrice ce interferează cu acestea.

Alte surse externe ale semnalelor electrice ce pot afecta calitatatea acestor semnale sînt: motoarele electrice, sistemele radio şi chiar lumina. În limbajul "rețelistic" acestea sînt denumite *interferențe electromagnetice* şi *interferențe radio*.

Spuneam că fiecare cablu în parte poate acționa ca o antenă. Dacă ne referim la UTP, sînt opt fire care se pot manifesta astfel. Ce se întîmplă de fapt? Fiecare fir din cablu absoarbe semnalele lectrice din celelalte fire ale cablului şi din sursele externe ale acestuia. Dacă "zgomotul' astfel rezultat este destul de ridicat, cartela de rețea nu va mai şti să facă diferență între acest zogmot şi semnalele care reprezintă date.

Problema interferenţelor este deosbit de importantă dacă ţinem cont de faptul că majoritatea LAN-urilor folosesc frecvenţe cuprinse între 1-100 megahertzi, bandă în care operează multe din posturile de radio FM sau semnalele TV.

Ideal ar fi ca înainte de proiectarea clădirii să se poarte discuţii cu cel care va realiza instalaţia electrică pentru a şti cu exactitate pe unde vor fi trase firele de alimentare cu electricitate. Se va încerca astfel ca liniile de comunicaţie ale reţelei să nu fie în apropierea (să nu se intersecteze) liniilor electrice. Pentru a evita neajunsurile create de acest tip de emisii, proiectanţii de materiale pentru reţea folosesc două tehnici: protejarea/izolarea şi anularea.

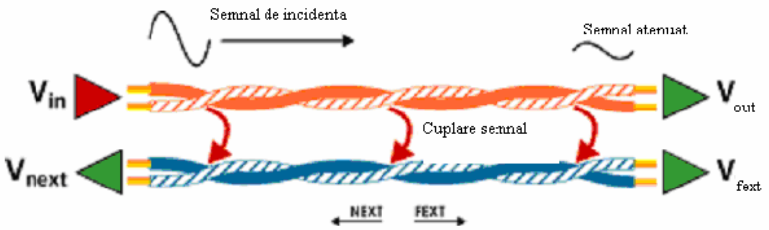
Protejarea presupune că fiecare pereche de fire sau grup de perechi este izolată de celelalte printr-un înveliş izolant. Acesta acţionează ca o barieră împotriva oricăror interferenţe. Dar aceasta nu este o soluţie tocmai viabilă, deoarece înfăşurarea unui material protector pe fiecare pereche de fire duce, pe de o parte, la creşterea diametrului cablului, iar pe de altă parte la creştere costului de producţie a acestuia.

Prin urmare, astăzi, cea mai folosită tehnică de protecţie împotriva interferenţelor o reprezintă anularea. Circulaţia curentului electric prin fire crează un cîmp magnetic circular în jurul acestora. Direcţia forţelor acestui cîmp este dată de direcţia de circulaţie curentului electric prin respectivul fir. Dacă două fire fac parte din acelaşi circuit, electronii se deplasează de la polul negativ sursă prin fir, către destinaţie. Apoi, de la destinaţie se întorc către polul pozitiv sursă. Aflîndu-se în apropiere, cîmpurile magnetice ale celor două fire se vor anula reciproc. Şi vor reuşi să anuleze şi orice alt cîmp magnetic din exteriorul circuitului din care fac parte. Înfăşurarea firelor între ele (torsadarea) poate creşte efectul anulării.(http://epics.aps.anl.gov/techpub/lsnotes/ls232/ls232.htm)

La trecerea curentului printr-un fir se crează un cîmp electromagnetic care poate să interfereze cu cel creat în firele adiacente. În cazul cablului UTP, fiecare pereche de fire este torsadată pentru a se anula cîmpurile electromagnetice create în cele două fire. Cu cît sînt mai des torsadate firele cu atît mai mult creşte efectul anulării şi rata reală de transfer a datelor prin respectivul cablu. Dacă firele nu sînt corect torsadate efectul este apariţia NEXT (near-end crostalk: încrucişare[23] la cel mai apropiat capăt).

Fenomenul este similar cu ce se întîmplă în cazul telefoniei cînd se ating două fire şi se aud simultan două convorbiri. În cazul reţelelor, fenomenul apare în momentul în care semnalul mai puternic dintr-o pereche de fire este recepţionat de o pereche adiacentă de fire. NEXT reprezintă acea porţiune a semnalului transmis care se cuplează electromagnetic cu semnalul recepţionat

Simplificat, lucrurile se prezintă astfel:



FEXT (far end crosstalk – încrucişare la capătul cel mai îndepărtat) este simalară cu NEXT, diferenţa constînd în faptul că încrucişarea este măsurată la capătul cel mai îndepărtat faţă de punctul de generare.

[23] Termenul exact este *diafonie*

Am insistat ceva mai mult asupra acestui subiect pentru că este bine de ştiut că folosirea cablurilor din categoria 5, a prizelor şi conectorilor de calitate nu conduce în mod automat la obținerea performanțelor unei rețele din această categorie.

Latența

În cadrul unei rețele, latența este de multe ori sinonimă întîrzierii: timpul necesar unui pachet să ajungă de la sursă la destinație. În practică, acest fenomen are mai multe surse.

Putem vorbi în primul rînd de *dispersia* caracteristică materialului din care este realizat un cablu. Este posibil ca 1 bit să interfereze atît cu precedentul cît şi cu următorul. Iar cînd se transmit milioane de biti, este posibil ca acest lucru să depăşească limitele normale în ceea ce priveşte timpul de transmisie.

Toate sistemele digitale au un ceas (sau o frecvență de tact la care lucrează), ceea ce înseamnă că pulsurile acestui ceas conduc la apariția unui eventiment sau altul: CPU să realizeze un anumit calcul, datele să fie scrise în memoria calculatorului, cartela de retea să transmită mai departe bitii pe care i-a recepționat...Dacă ceasul unei surse care transmite date nu este sincronizat cu cel al destinatarului se poate ajunge la distorsionarea în timp a transmisiei, ceea ce face ca bitii să ajungă la destinație mai tîrziu decît ar fi normal.

Latența, cunoscută şi sub denumirea de întîrziere, are două cauze principale. Prima, se datorează teoriei relativității. Prin firele metalice şi prin fibra optică, semnalele se propagă cu o viteză mai mică decît a rețelei ($2,3*10^8$ m/s, respectiv $2*10^8$ m/s). Prin urmare, pentru a traversa o anumită distanță, bitii au nevoie de un anumit timp. La aceasta se mai adaugă şi faptul că majoritatea componentelor electronice ale calculatorului induc latență (este vorba de fracțiuni de secundă!)

Dispersia poate fi "combătută" prin proiectarea corespunzătoare a cablurilor, reducerea lungimii acestora, şi folosirea impedanței corespunzătoare. În cazul fibrei optice, limitarea dispersiei presupune folosirea unei lumini laser cu o lungime de undă specifică.

În cazul distorisiunii, lucrurile sînt ceva mai complicate deoarece este vorba de sincronizări complexe la nivel hardware.

Latența se reduce cu ajutorul echipamentelor de rețea şi a protocoalelor specifice nivelurilor modelului OSI.

10.2 Coliziunea şi domenii de coliziune

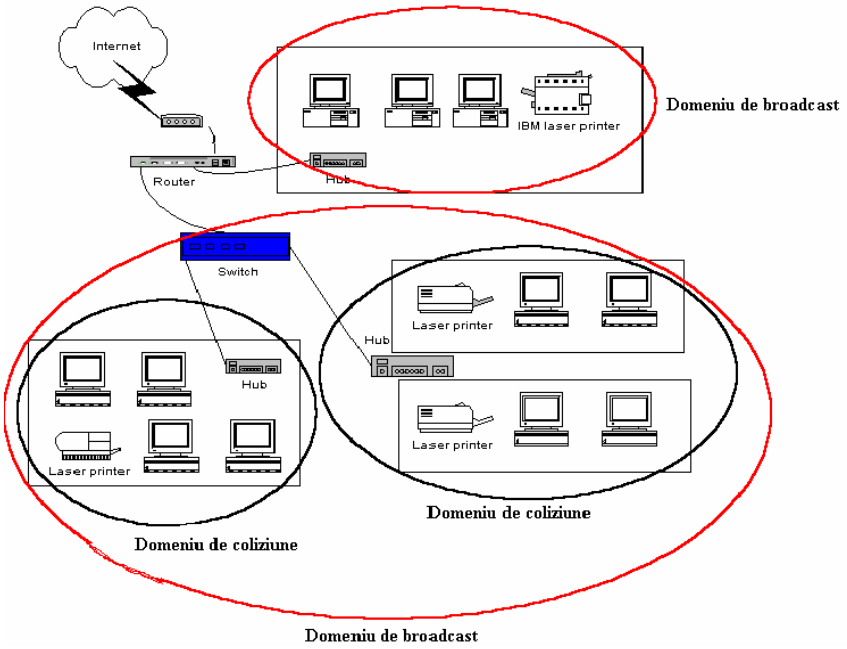
Coliziunea apare în momentul în care bitii transmişi de două calculatoare se întîlnesc pe acelaşi mediu de transmisie. Fizic, în cazul cablurilor bazate pe cupru acest lucru înseamnă un plus de voltaj în respectivul mediu, lucru care nu este permis în sistemele binare care sînt capabile să *înțeleagă* doar două valori ale voltajului.

Majoritatea tehnologiilor de rețea se confruntă cu această problemă. În unele cazuri, coliziunile sînt parte componentă a rețelei. Prea multe coliziuni însă pot conduce la încetinirea rețelei sau chiar la întreruperea funcționării acesteia. Dacă există reguli în baza cărora această problemă să poată fi controlată, lucrurile nu sînt deloc complicate.

Rețelele Ethernet folosesc mecanismul CSMA/CD prin care identifică coliziunile apărute la un moment dat.

Porțiunea de rețea în care pachetele transmise intră în coliziune se numeşte *domeniu de coliziune*. Acest fenomen apare în toate cazurile în care mediul de transmisie este partajat între mai multe calculatoare. Toate conexiunile care se fac prin intermediul dispozitivelor de nivel 1 fac parte dintr-un domeniu de coliziune.

Pachetele implicate într-o coliziune sînt distruse bit cu bit. Spuneam că pentru a se evita astfel de situații, rețeaua trebuie să dețină un mecanism (CSMA/CD, de exemplu) prin care să gestioneze conflictele care apar. În funcție de tehnologia folosită, numărul calculatoarelor care pot folosi împreună un mediu de transmisie (denumit *segment*) este limitat.



Despre huburi (repetoare) am spus că sînt dispozitivele care regenerează traficul unei rețele fără însă a filtra în vreun fel informațiile pe care le recepționeză/transmit. Informațiile primite de un port al unui hub sînt transmise tuturor celorlalte porturi. Prin urmare folosirea acestor dispozitive conduce la extinderea unui domeniu de coliziune.

Există o regulă care trebuie respectată în rețelele Ethernet: *numărul maxim de huburi între două calculatoare din rețea trebuie să fie 4*! Aceasta deoarece fiecare hub induce latență, iar depășirea acestui număr duce la creşterea numărului de coliziuni întîrziate. Coliziunea întîrziată apare după transmiterea primilor 64 bytes din cadru. Chipseturile plăcilor de rețea nu au capacitatea de a retransmite informațiile în urma apariției unei coliziuni întîrziate.

În practică această regulă este cunoscută sub denumirea **5-4-3-2-1:** cinci secțiuni de rețea, 4 repetoare, 3 segmente cu hosturi, 2 segemente de legătură şi un domeniu de coliziune.

Chiar dacă sînt dispozitive ieftine, repetoarele sînt principalele echipamente care contribuie la extinderea domeniilor de coliziune. Rezolvarea acestui neajuns constă în *segmentarea rețelei* cu ajutorul switch-urilor şi ruterelor.

Două sînt motivele pentru care vom dori să segmentăm o rețea:

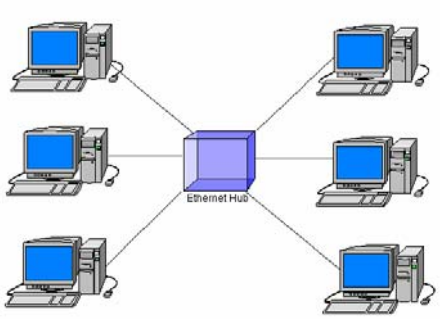
- izolarea traficului între segementele rețelei

- obținerea unei lățimi de bandă mai mari pentru utilizatori prin crearea unor domenii de coliziune reduse.

Fără segementare, o rețea mai mare decît cea destinată unui grup de lucru (de obicei maximum 10 calculatoare), va deveni foarte repede "inundată' de coliziuni, ba chiar virtual, ar ajunge în situația de a nu mai avea disponibilă lățime de bandă.

O rețea Ethernet care folosește switch-uri conduce la o topologie care se comportă ca și cînd ar exista doar două noduri în respectiva rețea: un nod sursă și un nod destinație. Aceste două noduri partajează, de exemplu, 10 Mbps între ele, ceea ce înseamnă întreaga lățime de bandă disponibilă pentru transmisia datelor.

Folosirea acestor echipamente permite rețelei să funcționeze mult mai eficient deoarece disponibilitatea lățimei de bandă se apropie de 100%.



În cazul unui switch, fiecare nod reprezintă o conexiune către un segment de rețea. Crearea acestor microsegmente de rețea conduce la eliminarea coliziunilor dintr-un domeniu. Fiecare nod este conectat direct la un port al switch-ului, sau la un segment care este conectat la un port. Prin aceasta se crează o conexiune pe 10 Mbps între fiecare nod și fiecare segment al switch-ului. Un calculator conectat direct la un port al switch-ului va reprezenta propriul său domeniu de coliziune și va avea acces la întreaga lățime de bandă: 10Mbps.

Switch-ul citește adresa sursă și/sau destinație a fiecărui cadru pe care îl primește. În funcție de informațiile acumulate va hotărî ce decizie trebuie luată: dacă trebuie comutat către un alt segment sau nu.

Dacă switchul este considerat un echipament pasiv ce acționează doar la nivelul legatură date, routerul acționează la nivelul rețea și ia decizii pe baza adresleor folosite de protocoalele acestui nivel. Acest lucru presupune examinarea adreselor destinație conținute în fiecare pachet și consultarea tabelelor de rutare cu privire la instrucțiunile ce trebuie urmate.

Cu ajutorul acestor echipamente se atinge cel mai ridicat nivel de segmentare datorită capacității lor de a decide cu exactitate unde trebuie să ajungă un pachet și ce cale trebuie să urmeze. Aceste acțiuni induc însă inevitabil latență.

Capitolul 11 Cablarea structurată a unei rețele de calculatoare

Ce reprezintă cablarea structurată? În cuvinte simple, reprezintă o arhitectură pentru comunicațiile prin cablu specificată de către comitetul EIA/TIA TR42[24] şi folosită în mod voluntar ca standard de către producătorii de echipamente.

Un sistem de cablare structurată include mediile de transmisie şi hardware—ul asociat cu scopul de a furniza o infrastructură de comunicații curpinzătoare. Această infrastructură nu trebuie să fie dependentă de un anumit dispozitiv. Mai mult, un sistem de cablare structurată începe din punctul în care se termină infrastructura furnizorului de servicii. Chiar dacă respectă aceeaşi metodologie, orice sistem de cablare structurată este unic în felul său din mai multe motive:

- structura arhitecturală a clădirii în care se relizează instalarea;
- produsele folosite;
- funcția îndeplinită de sistemul de comunicații;
- configurația echipamentelor;
- cerințele si limitările clientului.

11.1 Standardul ANSI/TIA/EIA 568

Elementele unui sistem de cablare structurată sînt:

- cablarea orizontală
- cablarea principală (coloana vertebrală a rețelei[25])
- zona de lucru (birourile)
- panoul pentru telecomunicații (rack sau dulapul în care sînt depozitate echipamentele de rețea)
- sala echipamentelor
- facilitățile de acces

Cablarea orizontală

Acest subsistem acoperă suprafața cuprinsă între priza din zona de lucru (birou) şi panoul pentru telecomunicații. Cablarea orizontală trebuie configurată într-o topologie stea, fiecare rack din zonele de lucru fiind conectat printr-o conexiune încrucişată la sala cu echipamente pentru telecomunicații.

Distanța maximă a unui segment va fi de 90 metri la care se adaugă cablurile din zona de lucru cu o lungime de maxim 10 metri. Pe lîngă cablurile din acest subsistem mai fac

[24] Electronics Industry Alliance/Telecommunications Industry Association – asociație care asigură în mod volutar standardele pentru interoperabilitatea echipamentelor de comunicație produse de mebrii săi.

25 Backbone – partea de rețea care sprijină traficul principal al rețelei

parte: conectorii pentru telecomunicații (prizele), conectorii şi cablurile din interiorul rackului.

Mediile de transmisie recunoscute de standard sînt:

- cablul UTP de 100 ohm
- cablul STP de 150 ohm
- fibra optică multimod (62.5/125 μm)

În fiecare zonă de lucru trebuie să existe o priză cu două module:

- un modul pentru cablu UTP cel puțin categoria 3, de 100 ohm
- un modul pentru unul din mediile de transmisie prezentate mai sus.

Cerințele legate de montaj trebuie să respecte specificațiile standardului ANSI/TIA/EIA-607.

Cablarea principală (backbone)

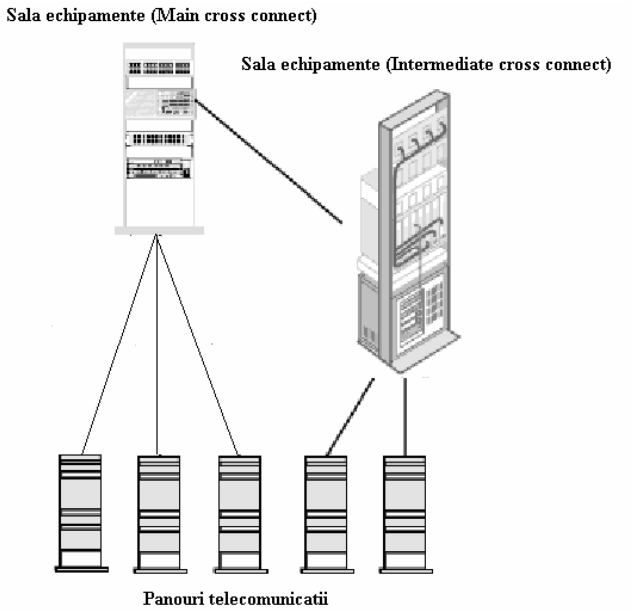
Backbone-ul rețelei suportă cantitatea cea mai mare de trafic şi interconectează între ele dulapurile cu echipamente, sălile cu echipamente sau facilitătile de acces.

În acest caz topologia este stea ierarhică şi poate fi completată în funcție de situație si cu topologii inel sau magistrală.

Distanțele maxime admise[26] pentru mediile de transmisie sînt:

Mediu de transmisie	Distanță maximă admisă pentru backbone
100 ohm UTP (24 or 22 AWG)	800 metri pentru voce, 90 metri pentru date*
150 ohm STP	90 metri pentru date
Fibră optică multimod 62.5/125 μm	2000 metri
Fibră optică single-mode 8.3/125 μm	3000 metri

[26] Aceste distanțe sînt dependente de aplicațiile implementate.



Patch cordurile nodului central şi cele din nodurile secundare nu ar trebui să depăşească 20 de metri. Cablurile pentru conectarea celorlalte echipamente nu ar trebui să depăşească 30 de metri.

Zona de lucru

Această componentă se întinde de la priza pentru comunicații până la stația de lucru sau alt echipament de lucru. Cablarea în acest caz trebuie să permită adaptarea rapidă la modificările care intervin: adăugarea unor stații noi, de exemplu.

Componentele acestui subsistem includ:

- stațiile de lucru, terminalele, telefoanele etc.
- Cablurile de prelungire (patch cablurile)
- Diferite adaptoare

Panoul pentru telecomunicații

Panoul pentru telecomunicații reprezintă zona în care sînt concentrate echipamentele prin care se interconecteză mediile de transmisie: terminatorii mecanici şi sistemul de cabluri cross sau backbone. Specificațiile tehnice ale acestui panou sînt definite prin standardul EIA/TIA 569A.

Sala echipamentelor

Specificațiile pentru proiectarea sălii echipamentelor sînt prezentate în standardul EIA/TIA 569A. O astfel de sala poate prezenta oricare din facilitățile unui panou pentru telecomunicații.

Facilitățile de acces

Această componentă se referă la punctul de interacțiune dintre cablurile din interiorul clădirii şi cele ale backbounului. Cerințele fizice sînt definite prin EIA/TIA 569A

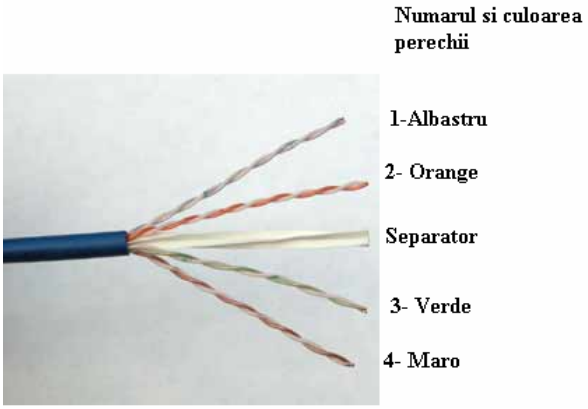
11.2 Instalarea cablurilor UTP

Chiar dacă am făcut o prezentare generală a acestui mediu de transmisie vom reveni acum cu problemele legate de instalarea sa. Cele mai multe probleme legate de rețele apar din cauza cablurilor: performanțele rețelei sînt strîns legate de cele ale cablurilor.

Majoritatea cutiilor în care este „împachetat" cablul sînt proiectate astfel încît desfăşuarea acestuia să fie cît mai uşoară. O astfel de cotie conține aproximativ 300 de metri de cablu pe care este marcată distanța din metru în metru (sau în picioare). DE fiacre dată cînd folosiți cablu dintr-o cutie notați pe aceasta numărul de metri utilizați. Veți şti astfel în orice moment cît cablu aveți la dispoziție pentru o lucrare.

Odată extras cablul din cutie marcați pe el portul sau locația în care îl amplasați.

Evitați astfel multe din neplăcerile ulterioare.
Perechile torsadate din cablul UTP au pe lîngă culori şi cifre după cum urmează:

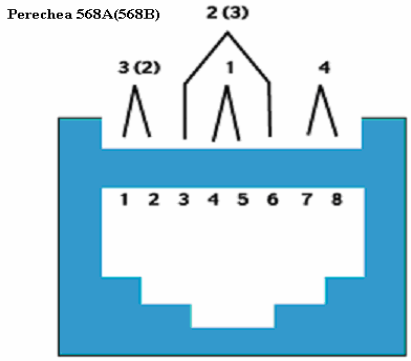


Este important de ştiut cifra corespunzătoare unei perechi pentru că un tester pentru cabluri raportează întotdeauna perechea cu probleme şi nu culoarea acesteia.

EIA/TIA 568B Pin#	Perechea#	Funcția	Culoarea firului	Ethernet 10/100 Base-T?	Ethernet 100 Base-TX şi1000 Base-T?
1	2	Transmite	Alb- portocaliu	Da	Da
2	2	Recepționează	Portocaliu	Da	Da
3	3	Transmite	Alb-verde	Da	Da
4	1	Nu se foloseşte	Albastru	Nu	Da
5	1	Nu se foloseşte	Alb-albastru	Nu	Da
6	3	Recepționează	Verde	Da	Da
7	4	Nu se foloseşte	Alb-maro	Nu	Da

8	4	Nu se foloseşte	Maro	Nu	Da

În jackul RJ 45, perechile sînt dispuse după cum urmează:



Secvența de culori pentrurealizarea conexiunilor este următoarea:

- Conexiune directă:

Folosită pentru a conecta un PC la un dispozitiv de rețea

- conexiune ineversată (crossover):

Folosită pentru a conecta două PC-uri între ele sau două dispozitive de rețea care nu au port special pentru o astfel de conexiune. Se observă că în acest caz perechea 2 este inversată cu perechea 3.

Cîteva sfaturi pentru instalarea cablurilor:

- toate componentele trebuie să fie de categoria 5e pentru a se atinge performanțele corespunzătoare;
- cablurile trebuie instalate fără a avea deformații sau noduri;
- cablurile nuj trebuie trase pe lîngă colțuri care formează unghiuri drepte.
- la instalările verticale este preferabil să se dea drumul cablului şi nu să fie tras pentru a se evita tensiunile;
- folosiți pe cît posibil cabluri de la producători certificați în acest sens.

Capitolul 12 Comutarea pachetelor în rețelele de calculatoare

Switching-ul (comutarea pachetelor) reprezintă o tehnologie ce contribuie la diminuarea congestiei în rețelele Ethernet, Token Ring şi FDDI prin reducerea traficului şi creşterea disponibilității lățimii de bandă. Astăzi switch-urile au ajuns să înlocuiască tot mai mult hub-urile deoarece este proiectat să funcționeze în infrastructura existentă deja în rețea fără a perturba traficul.

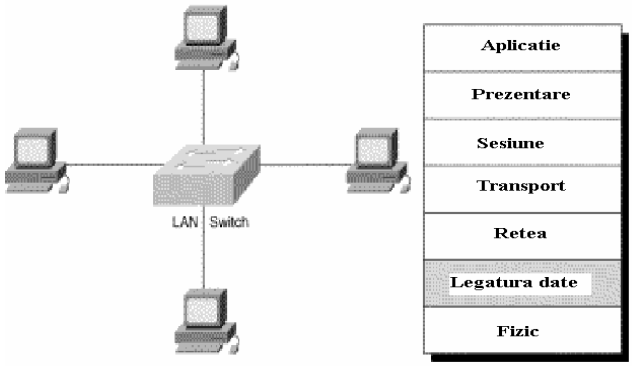
12.1 Funcţiile unui switch

Switch-urile au ajuns astăzi să fie considerate componenta fundamentală prin care se realizează segementarea celor mai multe reţele[27]. Permit utilizatorilor dintr-o reţea să transmită informaţii, prin acelaşi mediu, în acelaşi timp, fără a încetini traficul. Aşa cum routerele permit diferitor reţele să comunice unele cu altele, switch-urile permit diferitor noduri (nod = un punct de conexiune dintr-o reţea, de obicei un calculator) din reţea să comunice direct unele cu altele, într-o manieră eficientă.

Prin porturile sale (8, 16 sau 24) un switch împarte reţeaua în mai multe canale de comunicaţie. Aceste canale independente cres rawndamentul switchului în ceea ce priveşte lăţimea de bandă folosită. Switchurile mai simple sînt autoconfigurabile, prin urmare nu este nevoie de personal specializat pentru punerea lor în funcţiune.

În cuvinte simple, modul de funcţionare al unui switch este următorul: pentru un segment de reţea ataşat la un port al switchului, CSMA/CD va controla accesul al mediul de transmisie pentru respectivul segment. Dacă la respectivul port este ataşată o singură staţie de lucru nu este nevoie de nici un mecanism prin care să se controleze accesul al mediu. Switchl verifică adresele MAC sursă şi destinaţie ale cadrele pe care le recepţionează şi transmite respectivele cadre către porturile corespunzătorare.

Prin urmare, comutarea pachetelor la nivelul 2 OSI se bazează pe hardware sau altfel spus foloseşte adrese fizice(MAC).



Un switch îndeplineşte două funcţii principale:

- *Comutarea cadrelor*. Această funcţie are loc atunci când un cadru ajunge la switch dintr-un anumit mediu sau de pe un anumit port şi este transferat către un alt mediu/port
- *Gestionarea operaţiilor de comutare.* Switchul creează şi întreţine tabele de comutare sau de filtrarea folosind ASIC – Application Specific Integrated Circuits.

Fiecare switch folosit într-o reţea Ethernet induce latenţă. Un switch interpus între un server şi o staţie de lucru creşte timpul de transmisie cu 21 microsecunde. Un pachet de

[27] http://www.cisco.com/warp/public/473/lan-switch-cisco.shtml

1000 bytes are un timp de transmisie de 800 microsecunde. Dacă comutarea realizată de switch este de tip *store and foreward* latența indusă crește.

Tot la capitolul generalități menționăm cele două tipuri de switching: de nivel 2 sau de nivel 3. Diferența între aceste două tipuri de comutări constă în tipul informațiilor conținute în cadru: la nivel 2 se folosește adresa MAC, iar la nivel 3 informațiile nivelului 3.

Switchul nu analizează informațiile de nivel 3 conținute de un cadru ci doar adresa MAC a destinatarului. Dacă adresa este cunoscută, cadrul este transmis către interfața/portul corespunzătoare. Switchul construiește tabele cu adresele MAC corespunzătoare fiecărui port în parte.

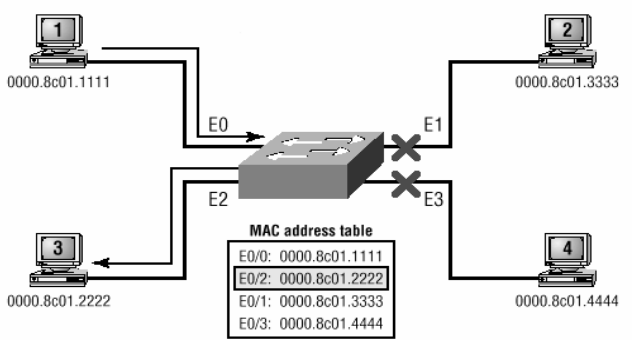
Dacă nu se cunoaște adresa destinatarului, cadrul este transmis către toate porturile (broadcast) pentru ca switchul să-i poată "învăța" destinația corectă. Când este reprimit cadrul, switchul adaugă adresa în tabela cu adrese MAC a portului respectiv.

Spuneam mai demult că adresele de nivel 2 sînt atribuite de către producătorii de astfel de echipamente și sînt alcătuite din două părți: codul producătorulu și un identificator unic al respectivului dispozitiv. Codul producătorului este stabilit de către IEEE în timp ce identificatorul este atribuit chiar de către producător.

Cu excepția SNA (Systems Network Architecture), utilizatorii nu au control asupra dreselor de nivel 2. În majoritatea rețelelor, administratorilor le revine sarcina de a atribui doar adrese de nivel 3. În acest caz putem spune că administratorii creează rețele locale ce se comportă ca un singur spațiu de adresare (blocul-strada-orașul-țara) .

Un switch Ethernet poate "învăța" adresa oricărui dispozitiv din rețea prin citirea adresei sursă conținută în fiecare pachet și notarea portului prin care cadrul a "intrat" în switch. Aceste adrese sînt memorate într-o bază de date. Adresele echipamentelor din rețea sînt memorate în mod dinamic, altfel spus, pe măsură ce apare un dispozitiv nou, adresa sa este citită, învățată și memorată într-o zonă de memorie (CAM-content addressable memory). Când switchul identifică o adresă pe care nu o regăsește în CAM, o memorează pentru o utilizare viitoare. În momentul memorării, adresa este "ștampilată" și cu data când a fost adăugată în CAM. Ori de câte ori o adresă este referită sau adăugată în CAM i se înregistrează și noua dată (inclusiv ora) la care a avut loc operațiunea. Adresele la care nu se face referire o anumită perioadă de timp sînt șterse din CAM. Prin acest mecanism, baza de date cu adresele MAC ale dispozitivelor din rețea este actualizată în mod constant.

12.2 Funcționare switchului



Pornind de la imaginea de mai sus vom încerca să vedem mai în detaliu cum funcționează un switch. La pornire, tabela cu adrese MAC nu conține nici o înregistrare. Când un calculator (1) transmite şi un port recepționează un cadru, switchul preia adresa MAC a calculatorului sursă si o plasează în tabela de filtrare împreună cu portul de unde a fost preluat. Având în vedere că destinatarul nu este cunoscut, switch-ul nu va avea de ales şi va trebui să "inunde" rețeaua cu acest cadru.

Dacă un calculator din rețea (3) răspunde şi trimite înapoi un cadru, switchul va prelua adresa sursă din acest cadru şi o va înregistra în baza de date cu adrese MAC în asociere cu portul de pe care a fost primit. Din acest moment, switchul va putea realiza o conexiune puncat-la-punct, cadrele fiind transmisie doar între cele două calculatoare.

De fiecare dată când un cadru este recepționat pe un anumit port, adresa MAC destinație va fi comparată cu înregistrările din baza de date a switchului. Dacă această adresă este cunoscută şi apare în baza de date, cadrul va fi transmis către portul corespunzător. În caz contrar, cadru este transmis broadcast (mai puțin portul pe care a fost recepționat), adresa calculatorului care răspunde la broadcast urmând a fi înregistrată în baza de date cu adrese MAC.

Dacă tot am pomenit de broadcast, trebuie făcută o precizare: cadrele broadcast şi multicast nu specifică adrese MAC destinatare. Adresa sursă va fi întotdeauna adresa MAC a dispozitivului care transmite cadrul, în timp ce adresa destinație poate fi o adresă broadcast (toți biții 1) sau una multicast (biții din porțiunea host au valoarea 1). Broadcastul va fi transmis către toate rețelele şi hosturile acestora în timp ce multicastul va fi transmis tuturor hosturilor unei anumite rețele.

Ce se întâmplă însă atunci când o legătură se întrerupe? Am putea spune că într-o rețea ar fi bine să existe legături redundante. Corect. Această rezolvare însă, mai mult încurcă decât ajută. Atât timp cît brodcastul este transmis tuturor rețelelor, furtuna broadcast sau altfel spus circulația mesajelor în buclă nu poate fi evitată. Ce se întâmplă în astfel de situații? Un calculator va primi copii ale aceluiaşi cadru, dar de la surse diferite, de pe segmente de rețea diferite. Tabela cu adrese MAC de pe switch nu mai ştie ce să înregistreze. Switchul nu ştie care cadru trebuie transmis în rețea deoarece tabela MAC este actualizată continuu. Această situație are o rezolvare şi se numeşte Spanning Tree Protocol.

Comutarea simetrică a pachetelor caracterizează o rețea în care switchul alocă lățimea de bandă în mod egal fiecărui port. Un switch simetric oferă deci conexiuni între porturi cu aceeaşi lățime de bandă: 10Mbps sau 100Mbps. Spre deosebire de acesta, un switch

asimetric oferă conexiuni între porturi cu lățimi de bandă diferite, cum ar fi o combinație 10-100Mbps.

Switch-urile asimetrice sunt folosite mai ales în cazul traficului generat de aplicații client-server. Într-un astfel de switch bufferingul memoriei trebuie să permită traficului de pe portul pe 100Mbps să fie transmis către portul pe 10Mbps fără să conducă la blocarea acestuia din urmă.

Un switch Ethernet foloseşte o tehnică buffering prin care memorează şi transmite pachetele către porturile corecte. Zona de memorie în care se păstrează datele ce trebuie transmise se numeşte buffer. Această zonă de memorie poate folosi două metode de transmise a pachetelor: *port-based memory buffering* sau *shared memory buffering* .

În bufferingul de tip port-based, pachetele sînt memorate într-o coadă de aşteptare în funcție de portul pe care au "intrat". Un pachet este transmis către destinație abia când toate pachetele dinaintea sa au fost transmise. Acest lucru face posibil ca un singur pachet să întârzie întreaga transmisie a pachetelor din memorie ca urmare a ocupării portului destinație (de exemplu.). Această întârziere apare chiar dacă alte pachete pot fi transmise către porturi care sînt libere.

În bufferingul de tip shared memory, switchul depozitează pachetele într-o zonă de memorie care este partajată între toate porturile. Memoria alocată unui port depinde de cît "cere" fiecare port la un moment dat, altfel spus este alocată în mod dinamic. Pachetele din buffer sînt asociate dinamic portului care transmite. Aceasta oferă posibilitatea ca un pachet să fie recepționat pe un port şi transmis pe un altul fără ca datele să mai "stea la coadă".

Switchul întreține o hartă a tuturor porturilor către care trebuie transmis un pachet. După ce transmisia pachetului s-a făcut cu succes, această hartă este ştearsă. Ca efect al memoriei partajate, dimensiunea unui pachet este limitată la dimensiunea bufferului şi nu doar la cea alocată unui anumit port. Ce înseamnă aceasta? Se pot transmite pachete mai mari dacă se distrug câteva pachete de dimensiuni reduse. Este o facilitate foarte importantă în cazul switch-urilor 10/100Mbps, în care un port pe 100 poate să transmită un pachet către un port pe 10 Mbps.

12.3Metode de comutare

Latența pachetelor într-un switch depinde în primul rând de modul în care se realizează comutarea acestora. Există trei astfel de metode.

Store-and-forward reprezintă cea mai cunoscută metodă de switching într-o rețea. Înainte de a fi transmis, cadrul este recepționat în totalitate: se citeşte adresa sursă şi/sau destinație şi se aplică anumite filtre, se calculează o cifră de verificare a redundanței. În timpul recepționării cadrului apare şi latența. Cu cît cadrul este mai mare cu atît este mai mare şi latența indusă ca urmare a timpului necesar citirii sale. Detectarea erorilor mai consumă şi ea ceva timp deoarece switchul trebuie să aştepte recepționarea întregului cadru. Dacă există erori, cadrul este distrus. Dacă este prea mic (mai puțin de 64 bytes) sau prea mare (mai mult de 1518 bytes) este de asemenea distrus. Dacă totul este bine, switchul caută adresa MAC destinație şi determină portul de ieşire.

Cut-through (în timp real) este cea de a doua metodă de comutare. Înainte de a aştepta recepționarea întregului cadru, switchul citeşte adresa destinație şi lansează transmiterea sa. De fapt switchul copie doar adresa destinație (primii 6 bytes de după preambul) în memorie şi caută această adresă în tabela sa pentru a determina care este portul de ieşire. Această tehnică reduce latența şi în plus nici nu detectează erorile aşa cum se întâmplă în store-and-forward.

Cut-through are două variante:

- *Fast-forward switching.* Această variantă induce cea mai mică latenţă deoarece un pachet este transmis imediat ce a fost identificată adresa destinaţie. Dezavantajul constă în faptul că sînt transmise mai departe şi pachetele ce conţin erori. Chiar dacă aceste situaţii nu apar în mod constant şi NIC-urile renunţă la cadrele ce conţin erori, apariţia unui trafic inutil în reţea nu este digerată de nici un administrator sârguincios. Switchul măsoară latenţa pornind de la primul bit recepţionat şi terminând cu primul transmis (FIFO)
- *Fragment-free switching*. Această variantă presupune filtrarea şi transmiterea numai a fragmentelor de pachete ce nu conţin erori. În mod obişnuit, un fragment ce ia naştere în urma unei coliziuni trebuie să fie mai mic de 64 bytes. Orice pachet mai mare decât această valoare este considerat valid şi prin urmare este recepţionat fără erori. În această situaţie, switchul aşteaptă până când un pachet recepţionat este validat şi apoi îl transmite către portul destinaţie.

Latenţa fiecăruia din modurile prezentate depinde de cum se realizează transmisia pachetelor. Cu cît este mai rapid modul de transmitere, cu atât este mai redusă latenţa. Dar pentru a se ajunge în această situaţie înseamnă că switchul alocă mai puţin timp pentru verificarea erorilor. Şi cu cît verificarea erorilor este mai redusă cu atât mai mult creşte numărul retransmisiilor.

12.3 Spanning Tree Protocol (STP)

Principala menire a acestui protocol este de a împiedica apariţia aşa ziselor circuite în buclă. STP monitorizează în mod constant reţeaua izolând buclele şi trecând anumite conexiuni în stand-by. În cadrul fiecărei VLAN rulează o instanţă separată a STP pentru a se asigura conformitatea topologiei reţelei cu standardele în vigoare.

STP prezintă cinci stări:

- Blocking-nu se transmite nici un cadru;
- Listening-nu se transmite nici un cadru, se “ascultă”
- Learning-nu se transmite nici un cadru, se învaţă adrese;
- Forwarding-se transmit cadre, se învaţă adrese
- Disabled-nu se transmit cadre

Vom mai reveni cu ceva amănunte în curând.

Capitolul 13 Reţele virtuale

Am văzut că folosirea switchului într-o reţea Ethernet are ca efect segmentarea acesteia în domenii de coliziune individuale. Numărul total de segmente ce se pot obţine prin folosirea unui switch alcătuieşte domeniul de broadcast. Acest lucru înseamnă că toate nodurile aparţinând tuturor segmentelor pot să vadă broadcastul transmis de un nod al unui segment.

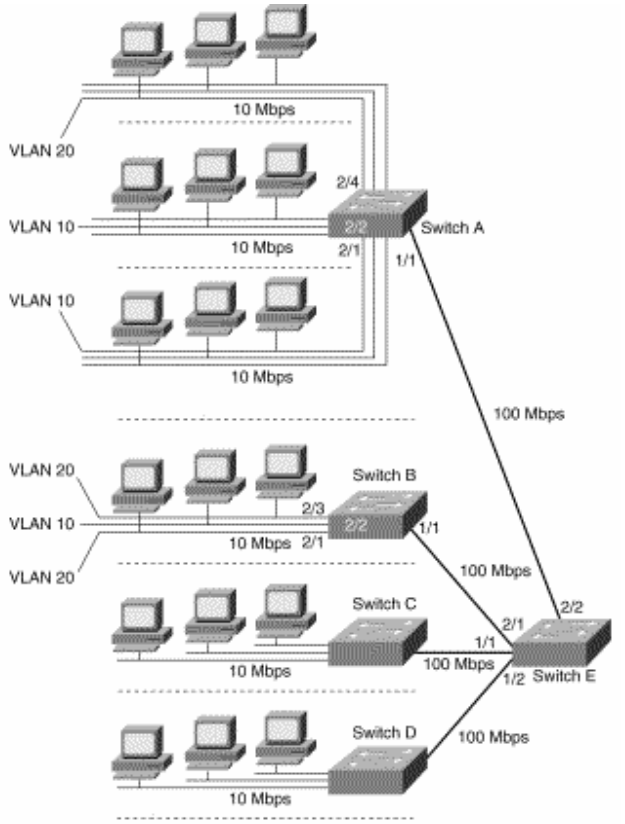
O rețea virtuală presupune gruparea logică a echipamentelor şi/sau utilizatorilor unei rețele fără a mai exista restricții legate de segmentul fizic din care fac parte. Altfel spus, o rețea virtuală reprezintă un domeniu de broadcast dintr-o rețea cu switchuri.

Cu ajutorul switchurilor se poate crea o singură rețea virtuală sau mai multe. În cel de al doilea caz, broadcastul unei astfel de rețele nu va fi „văzut" de către celelalte. Implementarea rețelelor virtuale permite administratorilor diminuarea domeniilor de broadcast şi creşetrea disponibilității lățimii de bandă.

13.1 Tipologia VLAN-urilor

Tehnologia pusă la dispoziție de VLANuri oferă posibilitatea grupării porturilor şi a utilizatorilor în grupuri logice. Dacă această grupare implică folosirea mai multor switchuri, VLANurile pot partaja o aceeaşi clădire, mai multe sau chiar WANuri. Pentru orice arhitectură VLAN, importantă este posibilitatea transferului de informații între switchuri şi rutere.

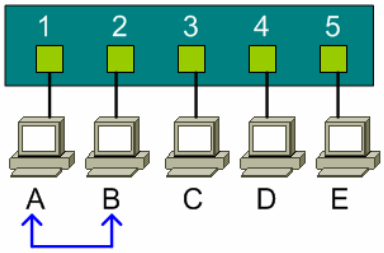
În mod tradițional, ruterul gestionează broadcastul şi procesează rutele pachetelore. Chiar dacă switchurile unei VLAN preiau o parte din aceste sarcini, ruterul rămîne vital pentru arhitectura oricărei rețele deoarece prin intermediul lor se pot interconecta VLANuri diferite.



Există mai multe modalităţi prin care se poate defini apartenenţa unui calculator la o VLAN, două fiind considerate mai importante: reţele virtuale bazate pe porturile switchului şi reţele virtuale ??????

13.1.1 Gruparea după porturi

Iniţial, multe implementări de VLAN-uri defineau apartenenţa la un VLAN prin intermediul unei grupări după porturile din switch. În plus, în majoritatea implementărilor iniţiale, VLAN-urile erau realizate pentru un singur switch.



Pentru a doua generaţie de implementări se poate forma un VLAN prin conectarea mai multor porturi de pe switch-uri diferite (Fig.).

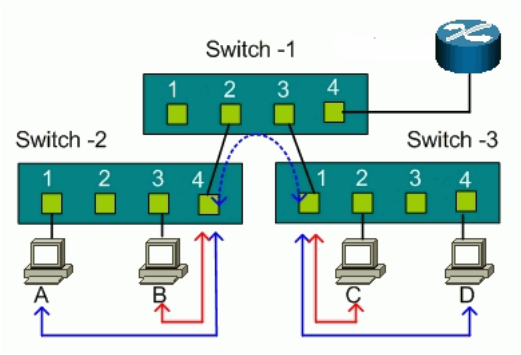


Figura nr. – Definire VLAN-uri după porturi

Gruparea după porturi este cea mai des întâlnită metodă, iar configurara este destul de uşor de realizat. Definirea VLAN-urilor doar pe baza porturilor nu permite unor VLAN-uri diferite să includă acelaşi segment fizic (sau port de switch). Însă, principala limitare în cazul grupării după porturi este că administratorul reţelei trebuie să reconfigureze VLAN-ul de fiecare dată când un utilizator este comutat de pe un port pe altul.

13.1.2 Gruparea după adresa MAC

Apartenenţa la un VLAN se realizează pe baza adresei MAC a staţiei. Switch-ul urmăreşte adresa MAC care aparţine fiecărui VLAN. Cum adresa MAC e o componentă a

plăcii de rețea, când se schimbă poziția unei stații nu mai este necesară nici o reconfigurare suplimentară pentru ca stația respectivă să rămână în același VLAN.

Gruparea într-un VLAN pe baza adreselor MAC are și avantaje și dezavantaje. Din moment ce adresele MAC sunt la nivelul 2 al modelului OSI și sunt "arse" în plăcile de rețea, VLAN-urile le permit administratorilor de rețea să schimbe poziția fizică a unui calculator în rețea, iar acesta să rămână în continuare în VLAN-ul din care făcea parte, în mod automat. În acest fel, un VLAN definit pe baza adreselor MAC poate fi considerat ca fiind "bazat pe utilizatori".

Unul din dezavantajele VLAN-urilor bazate pe adresa MAC e faptul că inițial, fiecare stație trebuie să fie adăugată manual la VLAN, mai precis fiecare adresă MAC de pe placa de rețea. După configurarea manuală inițială, urmărirea și repoziționarea stațiilor (și implicit a utilizatorilor) este automatizată, acest lucru fiind specific fiecărui echipament, în funcție de producătorul său. Acest dezavantaj al unei configurări manuale inițiale este cu adevărat problematic în cazul în care, în faza inițială e necesară repartizarea la VLAN-uri a unui număr foarte mare de utilizatori, de ordinul miilor, în rețelele foarte mari. Unii producători au venit cu soluții automatizate indermediare, prin care diferite instrumente software crează VLAN-uri, pe baza adreselor MAC, la nivel de subrețea.

VLAN-urile bazate pe adresele MAC ale stațiilor, implementate în medii distribuite, vor întâmpina probleme mari în ceea ce privește performanța rețelei, pentru că traficul dinspre și spre stații din VLAN-uri diferite va trece prin aceleași porturi. În plus, metoda primară de comunicare a informațiilor de apartenență la un VLAN bazat pe adrese MAC, va avea ca efect scăderea performanțelor pentru implementări la scară mare.

O altă limitare a grupării într-un VLAN pe baza adreselor MAC e atunci când un utilizator folosește un notebook și pentru a se putea conecta la rețea, conectează notebook-ul său la așa numitele stații de andocare (docking stations). Notebook-urile, bineînțeles că au o adresă MAC care rămâne aceeași, indiferent de locația utilizatorului. Stațiile de andocare au și ele o adresă MAC unică, dar cum utilizatorul își modifică mereu poziția, adresele MAC ale stațiilor de andocare vor fi mereu altele. Adresa MAC variind mereu, va fi imposibilă și rămânerea utilizatorului mereu în același VLAN.

13.1. 3 VLAN-urile bazate pe nivelul 3 OSI

VLAN-urile bazate pe informațiile de la nivelul 3 al modelului OSI folosesc tipul protocolului (în cazul în care suportă mai multe tipuri de protocoale) sau adresele de la nivelul rețea (adresa de subrețea pentru rețelele TCP/IP) pentru a putea determina apartenența la un anumit VLAN. Deși aceste VLAN-uri lucrează cu informații de la nivelul 3, aceasta nu e o funcție de routare și nu trebuie confundată cu routarea la nivelul rețea. Deși un switch analizează adresa IP a pachetului pentru a determina apartenența la un VLAN, nu se calculează nici o rută a pachetului. Noile switch-uri însă știu să interpreteze și informațiile încapsulate de protocoalele RIP sau OSPF. Cadrele ce traversează un switch sunt trimise pe un port sau altul pe baza algoritmului "Spanning Tree". Astfel, din perspectiva unui switch ce e folosit într-un VLAN bazat pe nivelul 3 OSI, acesta e vazut ca o rețea plată, ca și topologie, fără nici o ierarhie, ca și când ar lucra numai la nivelul legătură-date.

Odată facută diferența între VLAN-urile bazate pe informații de la nivelul rețea și routare, trebuie menționat că anumiți producători au dotat swhitch-urile pe care le realizează cu capacități de procesare a informațiilor de nivel 3. Aceste capacități au dus la manifestarea unor funcții ce sunt asociate în mod obișnuit cu routarea. Mai mult, switch-urile "multi layer" au funcții de forwarding ale pachetelor datorită chip-set-urilor ASIC

încorporate. Însă, indiferent unde ar fi localizate switch-urile sau routerele într-un VLAN, routarea este necesară pentru a asigura conectivitatea între VLAN-uri distincte.

Există câteva avantaje evidente în definirea VLAN-urilor la nivelul 3 OSI. În primul rând, ele asigură partiționarea pe baza tipului de protocol. Aceasta se poate dovedi o opțiune atractivă pentru administratorii de rețea care trebuie să implementeze o strategie de VLAN pe baza unui serviciu sau a unei aplicații. În al doilea rând, utilizatorii își pot modifica locația fizică fără a mai fi nevoie să re-configureze adresa de rețea a stației lor (acesta este un avanjat în special pentru cei a căror stații sunt configurate pe TCP/IP). În al treilea rând, definirea VLAN-urilor la nivelul 3 poate elimina nevoia pentru "frame tagging", necesară pentru a comunica între switch-uri informații referitoare la apartenența la un VLAN. Se reduce astfel traficul general.

Unul din dezavantajele definirii VLAN-urilor la nivelul 3 al modelului OSI (în comparație cu definirea pe baza adreselor MAC, sau pe baza porturilor) este performanța. Analiza adreselor de la nivelul rețea în pachete este o acțiune care necesită mai mult timp decât analiza adreselor MAC din frame-uri. Din acest motiv, switch-urile care folosesc informații de la nivelul rețea pentru definirea VLAN-urilor au o latență mai mare decât cele care folosesc informații de la nivelul legătură-date. Această diferențiere se întâlneşte la majoritatea producătorilor de switch-uri, însă nu este o regulă general valabilă.

VLAN-urile definite la nivelul rețea sunt eficiente în rețele bazate pe protocoalele TCP/IP, dar sunt mai puțin eficiente pentru rețelele bazate pe protocoalele IPX, DECnet sau AppleTalk, care nu presupun configurarea manuală la desktop. În plus, VLAN-urile definite la nivelul rețea al modelului OSI întâmpină dificultăți la procesarea pachetelor unor protocoale neroutabile, precum NetBIOS. Stațiile finale care rulează protocoale neroutabile nu pot fi diferențiate între ele, astfel că nu pot fi definite ca şi părți componente ale unui VLAN configurat la nivelul rețea.

13.2 Configurarea VLAN-urilor

Manual

Configurarea VLAN-ului se face doar manual. Atât setarea inițială, cât şi modificările şi repoziționările ulterioare de echipamente în rețea sunt controlate de către administratorul de rețea. Configurarea manuală are însă şi avantajul controlului total asupra rețelei. Însă, cu cât complexitatea rețelei şi dimensiunea acesteia creşte, cu atât devine mai dificilă întreținerea acesteia, astfel încât mentenanța manuală este aproape imposibilă.

În plus, administrarea manuală înlătură însăşi unul din avantajele pe care le presupune existența unui VLAN, şi anume eliminarea timpului necesar pentru administrarea schimbărilor şi mutărilor (deşi mutarea unui utilizator în interiorul unui VLAN este mai uşor de realizat decât mutarea unui utilizator dintr-o subrețea în alta).

Semi-Automat

Configurarea semi-automată se referă la existența posibilității de a automatiza fie configurarea inițială, fie modificărie şi mutările ulterioare, fie ambele. Automatizarea inițială e realizată de obicei printr-un set de instrumente care mapează VLAN-urile la subrețelele existente. Configurarea semi-automată poate de asemenea însemna că inițial, configurarea se realizează manual, urmând ca toate modificările şi mutările ulterioare să fie îndeplinite automat. Combinarea configurării inițiale automate cu urmărirea automatizată a schimbărilor tot presupune configurare semi-automată, pentru că administratorul are încă, la orice moment, posibilitatea de a interveni manual şi de a face orice schimbare.

Automat

Un sistem care are automatizată funcția de configurare a unui VLAN presupune că stațiile de lucru se conectează automat şi dinamic la VLAN, în funcție de aplicație, ID-ul utilizatorului, sau alte politici predefinite de către administrator.

13.3 Tipologia conexiunilor

Device-urile dintr-un VLAN pot fi conectate în trei moduri, în funcție capacitatea echipamentelor de a recunoaşte sau nu VLAN-urile. Un echipament care recunoaşte VLAN-urile este acel echipament care înțelege modul de apartenență (sau de definire) al unui VLAN şi cadrele din VLAN-uri.

Toate echipamentele conectate la o legătură trunchi (trunk link) trebuie să poată să identifice/recunoască VLAN-urile. Toate cadrele dintr-o legătură trunchi trebuie să aibă ataşat un header special. Toate aceste frame-uri speciale sunt denumite "tagged frames" (frame-uri de urmărire).

O legătură acces (access link) conectează un echipament ce nu este capabil să recunoască un VLAN de portul unui bridge/switch capabil să recunoască un VLAN. Toate frame-urile dintr-o legătură acces sunt "untagged frames" .Echipamentele incapabile să recunoască un VLAN pot fi reprezentate de un segment de LAN cu stații de lucru ce nu pot recunoaşte un VLAN sau poate fi un număr de segmente LAN ce conțin echipamente ce nu pot recunoaşte un VLAN.

Legătură hibridă (Hybrid Link) este o combinație între cele două tipuri anterioare de legături. Acesta este un tip de legătură în care ambele echipamente (care pot recunoaşte un VLAN şi care nu pot recunoaşte un VLAN) sunt conectate (Fig. 22). O legătură hibridă poate avea atât frame-uri "tagged" cât şi frame-uri "untagged", dar toate frame-urile pentru un anumit VLAN trebuie să fie ori "tagged" ori "untagged".

13.4 Procesarea frame-urilor

Un switch care primeşte date, determină cărui VLAN îi sunt destinate aceste date pe care le primeşte. Switch-ul stabileşte cărui VLAN le sunt destinate prin "tagging" implicit sau explicit. În tagging-ul implicit, un header de urmărire este adăugat la frame. Switch-ul ține de asemenea evidența, într-o bază de date, a tuturor stațiilor de lucru ce aparțin unui VLAN. Această bază de date este folosită pentru a determina portul pe care să fie trimise mai departe datele. În continuare va fi prezentat conținutul acestei baze de date. Totodată, va fi prezentat formatul şi continutul header-ului de urmărire (tag header) conform specificațiilor IEE 892.1Q.

Baza de date pentru filtrare (filtering database)

Informațiile cu privire la apartenența la un anumit VLAN sunt stocate într-o bază de date folosite pentru filtrare. În baza de date pentru filtrare se regăsesc următoarele tipuri de intrări:

- intrări statice (static entries)

Informațiile statice sunt adăugate, modificate şi şterse doar de administratorul de rețea. Intrările nu sunt şterse automat după o anumită perioadă de timp (nu funcționează mecanismul prezentat în primul capitol, şi care se numeşte îmbătrânire – ageing), ci trebuiesc şterse explicit de către administratorul de rețea. Sunt două tipuri de intrări statice

intrări statice de filtrare : acestea specifică, pentru fiecare port, dacă frame-urile trebuiesc trimise către o anumită adresă MAC sau un grup de adrese MAC, dintr-un anumit VLAN sau dacă frame-urile trebuiesc să fie distruse sau dacă ar trebui să urmeze o intrare dinamică

b) intrări statice de înregistrare : specifică dacă frame-urile ce vor fi trimise spre un anume VLAN vor fi urmărite sau nu şi care sunt porturile rezervate pentru acel VLAN.

intrări dinamice (dynamic entries)

Intrările dinamice sunt "învățate" de către switch şi nu pot fi create sau actualizate de către administratorul de rețea. Procesul de învățare observă portul pe care e primit un frame (cu o anumită adresă sursă şi ce provine dintr-un anumit VLAN) şi actualizează baza de date pentru filtrare.Intrarea este actualizată doar dacă sunt îndeplinite simultan următoarele trei condiții :

portul respectiv permite învățarea

adresa sursă este adresa unei stații de lucru şi nu o adresă de grup

există spațiu liber în baza de date

Intrările sunt şterse din baza de date de procesul de îmbătrânire (ageing process) după o anumită perioadă de timp (specificată excusiv de către administratorul de rețea, ce poate lua o valoare în intervalul 10 – 1.000.000 sec) doar dacă e posibilă o reconfigurare automată a bazei de date de filtrare atunci când se modifică topologia rețelei.

Există trei tipuri de intrări dinamice :

a) intrări dinamice de filtrare: acestea specifică dacă frame-urile destinate spre o anume adresă MAC de pe un anumit VLAN ar trebui trimise mai departe sau ar trebui distruse.

b) intrări dinamice de înregistrae a grupurilor: indică pentru fiecare port din switch dacă frame-urile destinate unui anumit grup de adrese MAC de pe un anumit VLAN ar trebui filtrate sau distruse. Aceste intrări sunt adăugate sau şterse folosind Group Multicast Registration Protocol (GMRP). Acest protocol permite traficului multicast să fie trimis pe un singur VLAN, fără a afecta traficul din celelalte VLAN-uri.

c) intrări dinamice de înregistrare: specifică care porturi sunt rezervate pentru un anume VLAN. Intrările sunt adăugate şi şterse folosind GARP VLAN Registration Protocol (GVRP), unde GARP reprezintă Protocolul de Înregistrare a Atributelor Generice (Generic Attribute Registration Protocol).

GVRP nu e folosit doar pentru a actualiza intrările dinamice de înregistrare, ci şi pentru a comunica informații către alte switch-uri ce pot recunoaşte VLAN-urile.

Pentru ca VLAN-urile să trimită datele către destinația corectă, toate switch-urile din VLAN ar trebui să dețină aceleaşi informații în bazele lor de date de filtrare. Switch-urile care pot recunoaşte VLAN-urile înregistrează şi trimit mai departe informațiile cu privire la apartenența la un VLA către toate porturile care fac parte din topologia activă a unui VLAN. Topologia a activă a unei rețele este stabilită când switch-urile sunt pornite sau atunci când apare o modificare în starea actuală a topologiei.

Topologia activă este determinată folosind un algoritm de tip Spanning Tree (arbore), care previne formarea de bucle de repetare în interiorul rețelei, realizând acest lucru prin scoaterea din funcțiune a porturilor. Odată ce s-a stabilit o topologie activă a rețelei (rețea ce poate conține mai multe VLAN-uri), switch-urile stabilesc o topologie activă pentru

fiecare VLAN în parte. Acest lucru poate avea ca rezultat topologii diferite pentru fiecare VLAN în parte, sau topologii similare pentru mai multe VLAN-uri. În oricare din cazuri, topologia VLAN-ului e un subset al topologiei active a rețelei pe ansamblu.

Căutarea (tagging-ul)

Când frame-urile sunt trimse în rețea, apare nevoia să se ştie cărui VLAN aparțin, astfel încât switch-urile să trimită aceste frame-uri numai către porturile din VLAN-urile din care fac parte, şi nu către toate porturile din switch. Această informație e adăugată la frame sub forma unui header de căutare ("tag frame"). În plus, header-ul de căutare :

- permite adăugarea de informații cu privire la ordinea priorității utilizatorilor,
- permite adăugarea de informații ce privesc controlul rutei
- indică formatul adresei MAC

Frame-urile la care a fost adăugat header-ul de căutare (tag header) sunt denumite frame-uri de căutare (tagged frames). Frame-urile de căutare (tagged frames) asigură transportul de informații cu privire la VLAN[28].

Frame-urile de căutare ce sunt trimise prin intermediul legăturilor trunchi (trunk links) sau legăturilor hibride (hybrid links) conțin un header de căutare (tag header).

[28] www.tele.sunyit.edu/virtual_lans.html

Capitolul 14: Privire generală asupra WAN

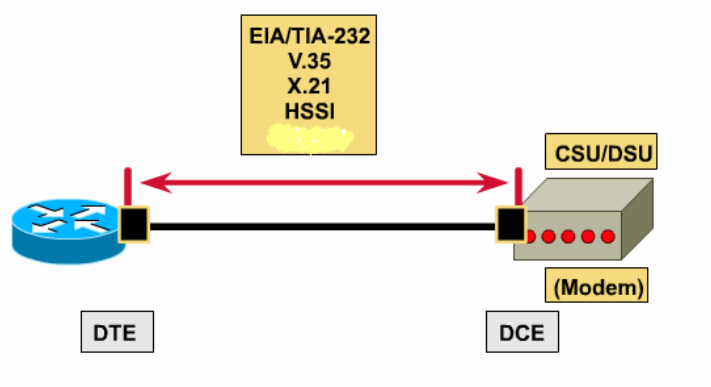
WAN operează la nivelul fizic şi legătură date din cadrul modelului de referinţă OSI şi interconectează LAN-uri distribuite în zone geografice distincte.

Caracteristicile principale ale WAN:

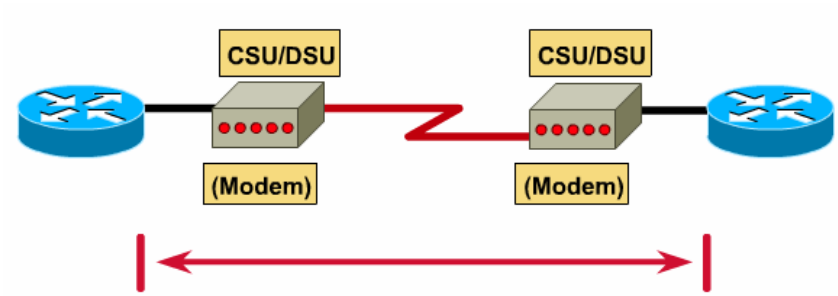
- Operează între LAN-uri;
- Conectează echipamente aflate în zone geografice distincte : rutere, switch-uri, modem-uri, servere

14.1Standardele WAN

Protocoalele de la nivelul fizic al WAN descriu conexiunile electrice, mecanice, operaţionale şi funcţionale pentru serviciile oferite de WAN. Cel mai adesea, aceste servicii sînt oferite de un provider (un operator de telefonie de exemplu).



Protocoalele nivelului legătură date descriu modul în care circulă cadru-urile între sisteme:



Standardele WAN descriu cerinţele nivelului fizic şi legătură date. La nivel fizic sînt descrise interfeţele dintre *data terminal equipment (DTE)* şi *data communication equipment (DCE)*. De obicei, DCE reprezintă furnizorul de servicii iar DTE reprezintă dispozitivul

ataşat. Standardele care descriu această interfață sînt: EIA/TIA-232, EIA/TIA-449, V.24, V.35, X.21, G.703, EIA-530 (în unele documentații apar precizate aceste standarde).

Iată care sînt şi standardele nivelului legătură date, standarde care se referă în principal la modul în care se realizează încapsularea datelor:

- High-Level Data Link Control (HDLC) – deşi este un standard IEEE, este posibil să nu ofere compatibilitate datorită manierei de implementare din partea producătorilor. HDLC suportă atât configurări point-to-point cît şi multipoint.
- Cadru Relay - foloseşte facilități digitale de înaltă calitate şi un framing fără mecanisme de corectare a erorilor. Aceasta înseamnă ca poate să transmită informațiile specifice nivelului 2 mult mai rapid decât alte protocoale WAN.
- Point-to-Point Protocol (PPP) – dezvoltat de către IETF, specifică un cîmp prin care se identifică protocolul nivelului rețea.
- Simple Data Link Control Protocol (SDLC) – dezvoltat de către IBM, este un protocol al nivelului legătură date pentru mediile System Network Architecture (SNA). Astăzi este înlocuit de mai versatilul HDLC.
- Serial Line Interface Protocol (SLIP) – unul din cele mai populare protocoale WAN, este astăzi înlocuit cu PPP
- Link Access Procedure Balanced (LAPB) – protocol al nivelului legătură date folosit de X.25 cu facilități extinse în ceea ce priveşte corectarea erorilor.
- Link Access Procedure D-channel (LAPD) – protocolul de nivel 2 folosit în cadrul recepției în tehnologia ISDN (despre aceasta cu altă ocazie)

14.2Tehnologiile WAN

Fără a intra prea multe amănunte, vom prezenta în continuare cele mai utilizate tehnologii în cadrul WAN.

Servicii cu comutarea circuitelor

- POTS (Plain Old Telephone Service*)* – deşi nu reprezintă un serviciu pentru comunicația de date între calculatoare, este inclus în această categorie din două motive: pe de o parte, multe din tehnologiile pe care le include fac astăzi parte din infrastructura aflată în continuă creştere; pe de altă parte, este considerat un model de încredere, uşor de folosit.
- Narrowband ISDN (Integrated Services Digital Network) – a reprezentat primul serviciu digital de tip dial-up. Utilizarea sa depinde de la o țară la alta, asigurând o lățime de bandă 128kbps-3Mbps.

Servicii cu comutarea pachetelor

- X.25 – deşi este o tehnologie veche, se mai foloseşte încă. Oferă siguranță în transmiterea datelor, dar lățimea de bandă este limitată la 2Mbps.
- Cadru Relay - este versiunea bazată pe comutarea pachetelor a Narrowband ISDN. A devenit cea mai populară tehnologie WAN asigurând o lățime de bandă maximă de 1,544Mbps.

Servicii cu comutarea celulelor

- ATM (Asynchronous Transfer Mode) – similară tehnologiei broadband ISDN, a devenit una din cele mai importante tehnologii WAN, cu o lățime de bandă de 622Mbps.
- SMDS (Switched Multimegabit Data Service) – este folosită momentan în cazul MAN-urilor şi asigură o lățime de bandă de 4,736Mbps.

Servicii digitale dedicate

- T1, T3, E1, E3 – Seriile T în Statele Unite ale Americii şi E în Europa, au devenit cele mai importante tehnologii WAN. Lățimile de bandă corespunzătoare sînt: T1 – 1,544 Mbps; T3 – 44,736 Mbps; E1 – 2,048 Mbps; E3 – 34,368 Mbps
- xDSL (familia de tehnologii Digital Subscriber Line) – Tehnologie WAN aflată în plină dezvoltare, dedicată în special home-user-ilor. Sunt incluse aici: *HDSL* -- high-bit-rate DSL; *SDSL* -- single-line DSL; *ADSL* -- asymmetric DSL; *VDSL* -- very-high-bit-rate DSL; *RADSL* -- rate adaptive DSL
- SONET (Synchronous Optical Network*)* – proiectată pentru medii bazate pe fibră optică, poate fi implementată şi în cazul firelor de cupru. Oferă lățimi de bandă de la 51,84Mbps la 9952Mpbs.

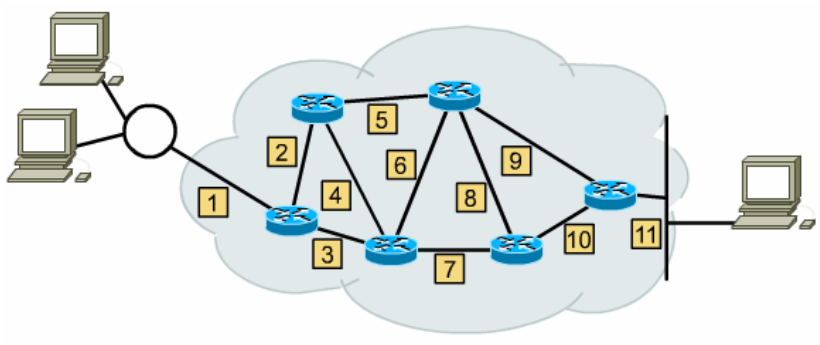
Alte tehnologii WAN

- Dial-up modem (switched analog) – limitată din punctul de vedere al lățimii de bandă (56kbps), foloseşte infrastructura telefonică existentă
- Cable modem – foloseşte ca mediu de transmisie cablul TV şi asigură o lățime de bandă maximă de 10Mbps (cam asta este tendința de pe piață noastră: marii operatori de pe piața TV- cazul Astral care a cumpărat recent DNT)
- Wireless – legăturile în aceste caz sînt de două tipuri: terestre sau prin satelit pentru utilizatorii “mobili”.

Capitolul 15: Privire generală asupra ruterelor

Aminteam pe la începutul acestui curs care sînt principalele componente ale unui calculator: procesorul, memoria, interfețele şi magistrala. Ruterul are aceleaşi componente şi prin urmare poate fi asemuit unui calculator care este dedicat executării unei sarcini mai speciale: rutarea informațiilor.

Ca şi în cazul calculatoarelor care au nevoie de un sistem de operare pentru a putea rula aplicații clasice, şi ruterul are nevoie de un astfel de sistem de operare care nu este nici Linux, nici Unix, nici măcar Windows. Sistemul de operare al ruterului se numeşte *Internetworking Operating Software (IOS)* şi foloseşte la rularea fişierelor de configurare ale ruterului. Prin intermediul acestor fişiere se controlează traficul dintre rutere, control care se bazează pe protocoalele de rutare prin intermediul cărora se direcționează protocoalele rutabile şi tabele de rutare.

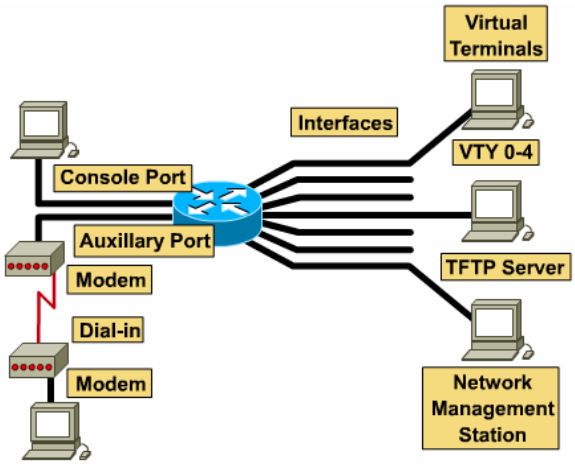


Pentru ca toate acestea să aibă loc, ruterul trebuie configurat aşa cum se configurează orice alt calculator (aproape).

15.1 Componentele ruterului şi moduri de configurare

Tehnic vorbind, ruterul este un calculator care selectează cel mai bun traseu (path) şi gestionează comutarea pachetelor între două rețele diferite. Un ruter poate fi configurat în mai multe moduri:

- de la un terminal în timpul instalării sale;
- via un modem, folosind un port auxiliar;
- de la un terminal virtual (VT0-4), după ce a fost instalat în rețea;
- de pe un server TFTP din rețea.



Să vedem în continuare care sînt componentele interne ale unui ruter:

- *RAM/DRAM* – reprezintă memoria în care se păstrează tabele de rutare, cahe-ul ARP, fast switching cache-ul, şi "coada" pe care o formează pachetele ce urmează a fi rutate. RAM oferă de asemenea memoria de lucru necesară (scuze dacă sună a pleonasm)fişierelor de configurare a ruterului. *În momentul*

întreruperii alimentării cu tensiune a ruterului sau restartării acestuia conținutul RAM se pierde!!!

- *NVRAM* – Memorie RAM nonvolatilă, păstrează fişierele startup/backup ale ruterului. După cum ii spune şi numele, nu se pierde dacă ruterul nu mai este alimentat cu tensiune.
- *Flash* – memorie ROM reprogramabilă, păstrează imaginea sistemului de operare şi microcodul acestuia. Permite actualizarea software-ului fără a se schimba chipurile procesorului. Poate păstra versiuni diferite ale sistemului de operare.
- *ROM* – conține instrumente de diagnosticare, un program bootstrap şi sistemul de operare. Dacă se doreşte up-grade la IOS trebuie înlocuite chip-urile microprocesorului.
- *Interfețe* – reprezintă conexiunile rețelei prin care pachetele intră şi ies din ruter. Pot fi amplasate direct pe placa de bază sau în module separate.

La o primă vedere cam aşa stau lucrurile. Cum însă RAM-ul este componenta cea mai importantă a ruterului trebuie să o privim cu mai multă atenție.

Această memorie reprezintă principala zonă de lucru a unui ruter. La pornire, în memoria ROM se execută un program bootstrap ce realizează anumite teste şi încarcă IOS-ul în memorie (din motive lesne de înțeles vom avea în vedere IOS-ul folosit de ruterele CISCO). Autoritatea executivă sau EXEC reprezintă una din componentele cele mai importante ale CISCO IOS. EXEc este cea care primeşte şi execută comenzile prin care administratorul configurează ruterul.

Memoria RAM este folosită si pentru păstrarea fişierului de configurare activă, a tabelelor ARP şi a tabelelor de rutare. Conținutul fişierului de configurare poate fi afişat prin intermediul unui terminal sau al unei console. O versiune a acestui fişier este păstrată şi în NVRAM. Acest fişier este încărcat în memorie la fiecare inițializare a ruterului. Despre informațiile conținute de acest fişier vom discuta câteva pagini mai încolo.

Spre deosebire de fişierul de configurare, imaginea IOS nu poate fi afişată pe ecranul unui terminal. Această imagine este de obicei executată în memoria RAM. Sistemul de operare este organizat în rutine ce gestionează sarcinile asociate diferitelor protocoale: transferul datelor, gestiunea tabelelor şi a bufferelor, actualizarea tabelelor de rutare, executarea comenzilor introduse etc.

Indiferent de modul în care este accesat (de la o consolă, prin intermediul unei sesiuni Telnet) un ruter se poate afla în moduri care să ofere funcționalități diverse:

- Modul ***user EXEC*** – este modul în care un utilizator poate să vizualizeze anumite informații despre ruter dar nu poate efectua nici o schimbare asupra stări acestuia.
- Modul ***privileged EXEC*** **–** este modul în care se pot introduce comenzi de testare şi debug, se poate examina în detaliu starea ruterului, se poate lucra asupra fişierelor de configurare şi se pot accesa celelalte moduri de configurare.
- Modul ***setup*** **–** este modul care permite crearea primei configurări a ruterului de la promptul unei console.
- Modul ***global configuration*** – este modul în care se obține accesul la comenzile de configurare.
- Modul ***RXBOOT*** – este un mod de întreținere care se foloseşte mai ales în cazul pierderii parolelor.
- Alte moduri de configurare: oferă opțiuni mai detaliate de configurare.

15.2 Ruterul: startup şi setup

Întrebare: de ce credeți că am prezentat comenzile de mai înainte şi abia acum discutăm despre setarea ruterului?

Inițializarea unui ruter presupune încărcarea programului bootstrap, a sistemului de operare şi a fişierului de configurare. Dacă ruterul nu găseşte un fişier de configurare trece automat în modul de lucru setup după care memorează în NVRAM o copie a noi configurări.

La pornire, ruterul execută o rutină de autodiagnoza, denumită **P**ower-**O**n-**S**elf-**Test.** În timpul acestei testări asupra modelelor hardware se execută programele de diagnoză din ROM pentru a se verifică gradul de operabilitate al CPU, memorie şi interfețelor. Dacă totul este în regulă, ruterul continuă cu inițializarea sistemului de operare.

După POST, pe măsură ce inițializarea ruterului are loc, au loc următoarele evenimente:

1. Se execută bootstrap-ul. Acesta este un program simplu prin care se încarcă instrucțiuni a căror efect este încărcarea în memorie a altor instrucțiuni care conduc spre modurile de configurare ale ruterului.

2. Se caută sistemul de operare după locația indicată în câmpul boot din registrul de configurare.

3. Se încarcă imaginea sistemului de operare. După ce devine operațional, sistemul de operare localizează componentele hard şi soft pe care le listează la consolă.

4. Fişierul de configurare salvat în NVRAM este încărcat în memoria principală şi executat linie cu linie. Comenzile de configurare pornesc procesul de rutare, atribuie adresele interfețelor, setează caracteristicile mediilor de transmisie etc.

5. Dacă în NVRAM nu există nici un fişier de configurare valid, sistemul de operare execută o rutină de interogare denumită *setup dialog.*

Setup-ul este folosit pentru a oferi ruterului o configurare minimă. Cu ajutorul comenzilor **show startup-config** şi **show running-config** se afişează cele două fişiere de configurare: activ respectiv backup. Dacă se doreşte ştergerea backup-ului fişierului de configurare din NVRAM se foloseşte comanda **erase startup-config**. Apoi, cu ajutorul comenzii **reload** se rebootează ruterul, acțiune ce are ca efect reluarea întregului proces de startup. Pentru a se ajunge în modul de lucru setup se tastează comanda cu acelaşi nume: **setup.**

Aminteam cu câteva rânduri mai înainte că scopul modului setup este de a oferi ruterului o configurație minimă în cazul în care nu găseşte informațiile necesare în alte surse. Majoritatea prompt-urilor din timpul configurării ruterului prin comanda **setup** oferă răspunsul default între paranteze pătrate. Setup-ul se poate anula în orice moment cu Ctrl+C, toate interfețele fiind oprite.

După terminarea configurării, aceasta va fi afişată pe ecran odată cu întrebarea daca doriți să folosii această configurare. Răspunsul "yes" la aceată întrebare va avea ca efect executarea instrucțiunilor de configurare şi salvarea lor în NVRAM. Dacă răspundeți cu "No" configurarea nu va fi salvată şi procesul va reîncepe.

După vizualizarea unui sumar al interfeței curente, pe ecran este afişat promptul corespunzător introducerii parametrilor globali ai ruterului. Aceşti parametri reprezintă valorile de configurare pe care le selectați.

Primul parametru global vă permite stabilirea unui nume de host pentru ruter. Acest nume va deveni parte a prompturilor IOS-ului de la CISCO pentru toate modurile de configurare. În timpul configurării inițiale, numele implicit al ruterului va fi afişat între paranteze pătrate, **[Router].**

Următorul parametru global ce va fi folosit se referă la parolele utilizate pentru protejarea ruterului. La început trebuie introdusă o parolă validă ("enable"). Algoritmul de criptare CISCO preia parola după ce a fost introdusă de la prompt ("Enter enable secret"). Astfel, oricine ar lista conținutul fişierului de configurare de pe ruter nu va putea vizualiza parola în clar.

În timpul setup-ului se recomandă ca "enable password" (parola)) să fie diferită de "enable secret word". Toate parolele sunt case sensitive şi pot conține caractere alfanumerice.

După ce se răspunde afirmativ la ultima întrebare din cadrul configurării inițiale, ruterul este gata de utilizare.

15.3 Configurarea ruterului

La startare, ruterul foloseşte următoarele informații din cadrul fişierului de configurare:

- Versiunea Cisco IOS-ului;
- Indentificarea ruterului;
- Locația(iile) fişierului folosit la bootare;
- Informații despre protocoale
- Configurarea interfețelor.

Dacă nu este disponibilă nici o configurare, se porneşte dialogul de configurare (setup).

Informațiile privind configurarea ruterului pot fi generate în mai multe moduri. În modul privilegiat EXEC se poate folosi comanda **configure** de la un terminal virtual sau o

consolă. Această variantă permite modificări asupra unei configurări existente. Aceeaşi comandă se poate folosi pentru a încărca o configurație de pe un server TFTP. Iată în continuare citeva din opțiunile disponibile la configurare.

- **configure terminal** – configurarea manuală a ruterului de la un terminal
- **configure memory** – încărcarea informațiilor de configurare din memoria NVRAM
- **copy tftp running-config** – încărcarea informațiilor de configurare de pe un server TFTP în memoria RAM.
- **show running-config** – afişarea configurării curente în RAM
- **`copy running-config startup-config`** – memorarea configurării curente din memoria RAM în NVRAM
- **`copy running-config tftp`** – memorarea configurării curente din memoria RAM pe un server TFTP
- **`show startup-config`** – afişarea configurării salvate în NVRAM
- **erase startup-config** – ştergerea conținutului NVRAM

15.4 Principalele comenzi ale IOS-ului

Ştiți care este una din cele mai arzătoare dorințe ale unui administrator de nivel 3? Să poată monitoriza starea ruterului/ruterelor în orice moment. Ruterele de la CISCO au o serie de comenzi ce permit administratorului să determine dacă acestea funcționează corect sau dacă au apărut probleme.

Comanda *enable* permite vizualizarea şi modificarea configurației ruterului în modul de lucru privilegiat:

```
Router>enable
```

Părăsirea acestui mod de lucru se face cu ajutorul comenzii *disable*. Părăsirea consolei de lucru se face prin comanda *logout* sau *exit.*

Configurarea ruterului se face cu ajutorul comenzii *config*.

```
Router#config
Configuring from terminal, memory, or network [terminal]?
```

Pentru a modifica setările unei interfețe se foloseşte comanda *interface*:

```
Router(config)#interface ?
  Async              Async interface
  BVI                Bridge-Group Virtual Interface
  Dialer             Dialer interface
  Ethernet           IEEE 802.3
  Group-Async        Async Group interface
  Lex                Lex interface
  Loopback           Loopback interface
  Null               Null interface
```

```
  Port-channel       Ethernet Channel of interfaces
  Serial             Serial
  Tunnel             Tunnel interface
  Virtual-Template   Virtual Template interface
 Virtual-TokenRing  Virtual TokenRing
```

Unele rutere nu suportă interfețe fastethernet, caz în care în loc de comanda *interface fastEthernet* trebuie să folosiți *interface ethernet:*

```
Router(config)#interface ethernet 0
Router(config-if)#
```

Pe fiecare interfață a unui ruter se pot crea sub-interfețe (interfețe virtuale):

```
Router(config-if)#interface ethernet 0.1
Router(config-subif)#
```

Dacă ruterul suportă FastEthernet comanda va *fi int f0/0.1*

Comanda *exit* vă întoarce în modul de configurare globală (`Router(config)#)`.

Pentru configurarea parolelor pentru modul de lucru utilizator se foloseşte comanda *line:*

```
Router(config)#line ?
  <0-6>    First Line number
  aux      Auxiliary line
  console  Primary terminal line
  vty      Virtual terminal
```

Pentru configurarea protocoalelor de rutare suportate de ruter se foloseşte comnda *router <nume_protocol>:*

```
Router(config)#router rip
Router(config-router)#
```

Ca şi în cazul switchurilor, helpul unui meniu/submeniu se activează cu ajutorul comenzii *?:*

```
Router#?
  Exec commands:
  access-enable    Create a temporary Access-List entry
  access-profile   Apply user-profile to interface
  access-template  Create a temporary Access-List entry
  bfe              For manual emergency modes setting
  clear            Reset functions
  clock            Manage the system clock
     configure        Enter configuration mode
```

Dacă vă interesează doar comenzile care încep cu o anumită literă tastați la prompt litera respectivă urmată de semnul ?

```
Router#c?
   clock  configure  copy
```

Să presupunem că vrem să configurăm data şi ora de pe ruter:

```
Router#clock ?
     set  Set the time and date
Router#clock set ?
     hh:mm:ss  Current Time
   Router#clock set 20:05:25 27 march 2003
```

Se verifică informațiile introduse:

```
Router#show clock
   20:05:25.526 UTC  27 March 2003
```

Comanda prin care se pot obține principalele informații despre un ruter este *show version*.

```
Router#sh version
Cisco Internetwork Operating System Software
IOS (tm) 2500 Software (C2500-JS-L), Version 12.0(8), RELEASE
SOFTWARE (fc1)
Copyright (c) 1986-1999 by cisco Systems, Inc.
Compiled Mon 29-Nov-99 14:52 by kpma
Image text-base: 0x03051C3C, data-base: 0x00001000

ROM: System Bootstrap, Version 11.0(10c)XB1, PLATFORM SPECIFIC
RELEASE SOFTWARE
(fc1)
BOOTFLASH: 3000 Bootstrap Software (IGS-BOOT-R), Version
11.0(10c)XB1, PLATFORM
SPECIFIC RELEASE SOFTWARE (fc1)

Router uptime is 2 hours, 26 minutes
System restarted by reload
System image file is "flash:/c2500-js-l_120-8.bin"

cisco 2500 (68030) processor (revision M) with 6144K/2048K bytes
of memory.
Processor board ID 17048803, with hardware revision 00000000
Bridging software.
X.25 software, Version 3.0.0.
SuperLAT software (copyright 1990 by Meridian Technology Corp).
TN3270 Emulation software.
1 Ethernet/IEEE 802.3 interface(s)
2 Serial network interface(s)
32K bytes of non-volatile configuration memory.
16384K bytes of processor board System flash (Read ONLY)

   Configuration register is 0x2102
```

Pentru a vizualiza informațiile despre configurația care rulează pe ruter se folosește *show running-config* sau *show startup-config*

```
Router#sh run
Current configuration:
!
version 12.0
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname Router
!
!
!
ip subnet-zero
!
interface Ethernet0
 no ip address
 no ip directed-broadcast
 shutdown
!
interface Serial0
 no ip address
 no ip directed-broadcast
 shutdown
!
interface Serial1
 no ip address
 no ip directed-broadcast
 shutdown
```

Configurația folosită de ruter la următoarea startare se afișează cu *show startup-config*

```
Using 781 out of 32762 bytes
!
version 11.2
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname Router
!
!
!
ip subnet-zero
!
interface Ethernet0
 no ip address
 no ip directed-broadcast
 shutdown
!
!
```

```
no ip classless
!
!
!
line con 0
line aux 0
line vty 0 4
!
end
```

Această configurație poate fi ştearsă de pe ruter: *erase startup-config*

```
Router#erase startup-config
Erasing the nvram filesystem will remove all files! Continue?
[confirm]
```

Se pot activa două parole pentru modul de configurare global

```
Router#config
Configuring from terminal, memory, or network [terminal]?t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#enable ?
  last-resort  Define enable action if no TACACS servers respond
  password     Assign the privileged level password
  secret       Assign the privileged level secret
  use-tacacs   Use TACACS to check enable passwords
Router(config)#enable secret ……(parola voastra)
Router(config)#enable  password  ……(cuvintul  folosit  ca  şi
cheie de criptare)
```

Pentru modul de lucru utilizator parola se va seta folosind comanda *line*

```
Router(config)#line ?
  <0-6>    First Line number
  aux      Auxiliary line
  console  Primary terminal line
  vty      Virtual terminal
```

aux – se foloseşte pentru parola utilizator pentru portul auxiliar de configurare

vty – setează o parolă Telnet pentru conexiunea la ruter. Dacă nu se stează o astfel de parolă telnetul nu poate fi folosit pentru conectarea la ruter

console – setează parola pentru configurarea ruterului de la consolă

Pentru configurarea parolei modului utilizator se alege linia dorită şi opțiunea *login* sau *no login* prin care va spune ruterului cum să se comporte la autentificare (cere sau nu parolă).

```
Router(config)#line aux 0
Router(config-line)#login
```

```
Router(config-line)#password ......(parola)
```

Pentru a seta parola consolei folosiți comanda *line console 0*.

```
Router(config)#line console ?
  <0-0>  First Line number
Router(config)#line console 0
Router(config-line)#password....(parola)
```

Am ales line 0 pentru că există doar un port pentru configurare de la consolă.

Pentru configurarea consolei există cîteva comenzi opționale care este bine să nu fie trecute cu vederea:

Exec-timeout 0 0 – intervalul de timp cît se poate lucra de la consolă

Logging synchronous - inhibă afişarea mesajelor pop-up ale consolei în timp ce lucrați

```
Router(config)#line con 0
Router(config-line)#exec-timeout ?
  <0-35791>  Timeout in minutes
Router(config-line)#exec-timeout 0 ?
  <0-2147483>  Timeout in seconds
  <cr>
Router(config-line)#exec-timeout 0 0   (lucru fără timeout)
Router(config-line)#logging synchronous
```

Luînd în calcul toate lucrurile rele ce s epot întîmpla vom seta şi o parolă pentru o sesiune Telnet.

Se accesează modul de lucru utilizator cu ajutorul comenzii *line vty*. Pentru a vedea cîte linii suportă versiunea IOS de pe ruter folosiți *line vty 0 ?*

```
Router(config)#line vty 0 ?
  <1-4>  Last Line number
  <cr>
Router(config)#line vty 0 4
Router(config-line)#login
Router(config-line)#password.....
```

Dacă încercați să vă conectați prin Telent la un ruter care nu are setată parolă pentru a o astfel de conexiune, primiți un mesaj de avertizare în acest sens. Se poate trece de acest neajuns folosind comanda *no login*, ceea ce va permite conexiune fără parole. Nu este înă sigur!!!

Pentru criptarea parolelor s eva folosi *sevice password-encryption:*

```
Router#config t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#service password-encryption
Router(config)#enable password adim
Router(config)#line vty 0 197
Router(config-line)#login
```

```
Router(config-line)#password adim1
Router(config-line)#line con 0
Router(config-line)#login
Router(config-line)#password adim2
Router(config-line)#line aux 0
Router(config-line)#login
Router(config-line)#password adim3
Router(config-line)#exit
Router(config)#no service password-encryption
Router(config)#^Z
%SYS-5-CONFIG_I: Configured from console by console
Router#sh run
Current configuration:
!
version 12.0
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname Router
!
!
enable password 7 7C4I0Y9F
!
ip subnet-zero
!
interface Ethernet0
 no ip address
 no ip directed-broadcast
 shutdown
!
interface Serial0
 no ip address
 no ip directed-broadcast
 shutdown
!
interface Serial1
 no ip address
 no ip directed-broadcast
 shutdown
line con 0
  password 7 7N9W4L5Z5D
  login
line aux 0
  password 7 7V7X8P9I0A
line vty 0 4
line vty 0 197
  password 7 6X5S2Y5Q5J
  login
```

Se poate salva manual configurația din DRAM în NVRAM folosind comanda *copy running-config startup-config* sau..

`Router#copy run start` – în acest moment fişierul de configurare creat va fi plasat în NVRAM de unde va fi folosit la următoarea bootare a ruterulrui

Bannerele reprezint[informații ce vor fi afişate de către ruter atunci cînd un administrator se loghează.

```
Router(config)#banner ?
  LINE      c banner-text c, where 'c' is a delimiting character
  exec      Set EXEC process creation banner
  incoming  Set incoming terminal line banner
  login     Set login banner
  motd      Set Message of the Day banner
```

Comanda *motd* este cea mai folosită deoarece afişează un mesaj oricărei persoane care se conectează la ruter prin Telent, port auxiliar sau consolă.

```
Router(config)#banner motd ?
  LINE  # banner-text #, where '#' is a delimiting character
Router(config)#banner motd #
Enter TEXT message.  End with the character '#'.
Daca nu sinteti autorizat sa folositi acest ruter, va rugam sa va
deconectati#
Router(config)#exit
Router#exit
Router Con0 is now available

Press RETURN to get started.




Daca nu sinteti autorizat sa folositi acest ruter, va rugam sa va
deconectati


Router>
```

În modul de configurare global tastați *interface ?*

```
Router(config)#interface ?
  Async              Async interface
  BVI                Bridge-Group Virtual Interface
  Dialer             Dialer interface
  Ethernet           IEEE 802.3
  Group-Async        Async Group interface
  Lex                Lex interface
  Loopback           Loopback interface
  Null               Null interface
  Port-channel       Ethernet Channel of interfaces
  Serial             Serial
  Tunnel             Tunnel interface
  Virtual-Template   Virtual Template interface
  Virtual-TokenRing  Virtual TokenRing
```

```
Router(config)#interface
```

Vizualizarea interfețelor seriale:

```
Router(config)#interface serial ?
  <0-1>  Serial interface number
```

Comanda de mai sus indică că acest model de ruter are două interfețe seriale.

```
Router(config)#interface Ethernet ?
  <0-0>  Ethernet interface number
```

Comanda de mai sus indică că ruterul folosit are o singură interfață Ethernet. Pentru configurarea acesteia se folosește următoarea comandă:

```
Router(config)#interface ethernet 0
Router(config-if)#
```

Pentru ruterele care suportă şi sloturi comanda de configgurare a interfeței este similară cu cea a switchurilor: *interface <tip interfață> <număr slot/port>*

Înainte de a putea fi efectiv configurată o interfață trebuie activată. Activarea/dezactivarea unei interfețe se face cu ajutorul comenzilor *shutdown/no shutdown*. Dacă o interfață este dezactivată acest lucru va fi afişat dacă folosiți comanda *show interface.* La pornirea unui ruter, interfețele sînt implict dezactivate.

```
Router#sh int e0
Ethernet0 is administratively down, line protocol is down
  Hardware   is   Lance,   address   is   0010.7b81.65e9   (bia
0010.7b81.65e9)
  Description:
  Internet address is 0.0.0.0
  MTU 1500 bytes, BW 10000 Kbit, DLY 1000 usec, rely 252/255,
load 1/255
  Encapsulation ARPA, loopback not set, keepalive set (10 sec)
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input never, output 02:11:34, output hang never
  Last clearing of "show interface" counters never
  Queueing strategy: fifo
  Output queue 0/40, 0 drops; input queue 0/75, 0 drops
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     0 packets input, 0 bytes, 0 no buffer
     Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
```

Activarea interfeței:

```
Router(config)#int e0
Router(config-if)#no shutdown
```

```
%LINEPROTO-5-UPDOWN: Line protocol on Interface Ethernet0,
changed state to up
%LINK-3-UPDOWN: Interface Ethernet0, changed state to up
Router(config-if)#exit
Router(config)#exit
Router#sh int e0
Ethernet0 is up, line protocol is up
  Hardware is Lance, address is 0010.7b81.65e9 (bia
0010.7b81.65e9)
  Description:
  Internet address is 0.0.0.0
  MTU 1500 bytes, BW 10000 Kbit, DLY 1000 usec, rely 252/255,
load 1/255
  Encapsulation ARPA, loopback not set, keepalive set (10 sec)
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input never, output 02:11:34, output hang never
  Last clearing of "show interface" counters never
  Queueing strategy: fifo
  Output queue 0/40, 0 drops; input queue 0/75, 0 drops
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     0 packets input, 0 bytes, 0 no buffer
     Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
```

Pe interfața E0 se va configura adresa IP 172.16.10.2/24

```
Router(config)#int e0
Router(config-if)#ip address 172.16.10.2 255.255.255.0
Router(config-if)#no shut
```

Pe o interfață se poate seta o a doua adresă de subrețea:

```
Router(config-if)#ip address 172.16.20.2 255.255.255.0 secondary
```

Pentru interfețele seriale trebuie configurată conexiunea DCE pentru ruterele care au o astfel de conexiune activă (2501e:

```
Router(config)# int s0
Router(config-if)#clock rate ?
Speed (bits per second)
  1200
  2400
  4800
  9600
  19200
  38400
  56000
  64000
  72000
  125000
  148000
  250000
```

```
  500000
  800000
  1000000
  1300000
  2000000
  4000000

  <300-8000000>    Choose clockrate from list above
```

Veţi observa că interfaţa S0 nu are conexiune la DCE şi prin urmare veţi obţine un mesaj de eroare.

Lăţimea de bandă de pe intefeţele seriale este folosită de către protocoalele de rutare (IGRP, EIGRP, OSPF) pentru a calcula cel mai bun transport către un host din reţea. În cazul protocolului RIP (mai bătrîn) lăţimea de bandă a unei astfel de interfeţe nu prezintă importanţă.

```
Router(config-if)#bandwidth ?
  <1-10000000>  Bandwidth in kilobits
Router(config-if)#bandwidth 64
```

Deşi am ajuns la configurarea intefeţeleor trebuie să facem un salt înapoi pentru că am uitat un lucru destul de important: nominalizarea ruterului:

```
Router(config)#hostname 2501A
```

Pentru a documenta setările ruterului este bine ca pentru fiecare interfaţă să se facă o scurtă descriere de genul:

```
2501A(config)#int e0
2501A(config-if)#description Retea vinzari
2501A(config-if)#int s0
2501A(config-if)#desc wan la Bucuresti circuit:6fdda4321
2501A(config-if)#exit
2501A(config)#exit
2501A#sh run
Current configuration:
!
version 12.0
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname 2501A
!
!
!
ip subnet-zero
!
interface Ethernet0
 description retea vinzari
```

```
 no ip address
 no ip directed-broadcast
 shutdown
!
interface Serial0
 description wan la bucuresti circuit:6fdda4321
 bandwidth 64
 no ip address
 no ip directed-broadcast
 shutdown
!
interface Serial1
 no ip address
 no ip directed-broadcast
 shutdown
```

Sau cu ajutorul comenzii *show interface*:

```
2501A#sh int e0
Ethernet0 is administratively down, line protocol is down
  Hardware   is   Lance,   address   is   0010.7b81.65e9   (bia
0010.7b81.65e9)
  Description: retea vinzari
  Internet address is 0.0.0.0
  MTU 1500 bytes, BW 10000 Kbit, DLY 1000 usec, rely 252/255,
load 1/255
  Encapsulation ARPA, loopback not set, keepalive set (10 sec)
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input never, output 02:11:34, output hang never
  Last clearing of "show interface" counters never
  Queueing strategy: fifo
  Output queue 0/40, 0 drops; input queue 0/75, 0 drops
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     0 packets input, 0 bytes, 0 no buffer
     Received 0 broadcasts, 0 runts, 0 giants, 0 throttles

Serial0 is administratively down, line protocol is down
  Hardware is HD64570
  Description: wan la bucuresti circuit:6fdda4321
  Internet address is 0.0.0.0
  MTU 1500 bytes, BW 1544 Kbit, DLY 20000 usec, rely 255/255,
load 1/255
  Encapsulation HDLC, loopback not set, keepalive set (10 sec)
  Last input never, output never, output hang never
  Last clearing of "show interface" counters never
  Queueing strategy: fifo
  Output queue 0/40, 0 drops; input queue 0/75, 0 drops
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     0 packets input, 0 bytes, 0 no buffer
     Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0
abort
```

Paza bună trece primejdia. Prin urmare este bine să verificăm din cînd în cînd setările făcute (ping, traceroute, telnet, show interface):

```
2501A#ping ?
  WORD       Ping destination address or hostname
  apollo     Apollo echo
  appletalk  Appletalk echo
  clns       CLNS echo
  decnet     DECnet echo
  ip         IP echo
  ipx        Novell/IPX echo
  vines      Vines echo
  xns        XNS echo
  <cr>

2501A#traceroute ?
  WORD       Trace route to destination address or hostname
  appletalk  AppleTalk Trace
  clns       ISO CLNS Trace
  ip         IP Trace
  ipx        IPX Trace
  oldvines   Vines Trace (Cisco)
  vines      Vines Trace (Banyan)
  <cr>

2501A#telnet ?
  WORD  IP address or hostname of a remote system
  <cr>
```

15.5 Accesul la alte rutere

Este posibil ca în acest moment să vă întrebaţi ce şi cum se întâmplă cu ruterele din imediata apropiere a ruterului pe care îl administraţi voi.

Cisco Discovery Protocol (CDP) fumizează o comandă *show* proprietară, comandă ce permite administratorului să acceseze informaţii despre cum sînt configurate ruterele care sînt conectate direct la ruterul administrat. CDP rulează la nivelul legătură date motiv pentru care echipamentele care lucrează cu alte protocoale de nivel 3 pot afla unele de altele (vă amintiţi că adresele nivelului legătură date sînt similare adreselor MAC).

Când un ruter pe care rulează IOS (Release 10.3 sau mai vechi) bootează, în mod automat se startează şi CDP. Acesta permite ruterului să detecteze ruterele aflate în vecinătatea sa pe care rulează CDP.

Administratorul de reţea va folosi comanda **show cdp neighbors** pentru a afişa actualizările făcute de CDP pe ruter, sau altfel spus despre reţelele conectate direct la respectivul ruter. Orice ruter pe care rulează CDP schimbă cu "vecinii" săi informaţii cu privire la protocoalele cu care lucrează.

Chiar dacă CDP rulează implicit la pornirea ruterului, administratorul trebuie să activeze explicit acest protocol pentru fiecare din interfeţele ruterului, folosind comanda

cdp enable. Ruterul păstrează informațiile pe care le primeşte de la "vecini" într-o zonă de memorie cache. Dacă primeşte un cadru care indică schimbarea unei informații cu privire la respectivul vecin, ruterul va înlocui informațiile vechi cu cele noi.

Comanda **show cdp interface** afişează valorile pentru timerii CDP, starea interfeței şi tipul încapsulării folosită de CDP. Valorile implicite ale timerilor sînt setate la 60 secunde pentru frecvența actualizărilor, respectiv 180 secunde pentru durata de viață a intrărilor CDP. Dacă ruterul primeşte o actualizare mai devreme decât este stabilit prin timer, sau dacă durata de viață a expirat, renunță la vechile informații.

Cu ajutorul comenzii **show cdp entry {nume ruter}** se vor conținutul memorie cache aferente unei singure intrări CDP inclusiv toate adresele de nivel 3 prezente în ruterul specificat.

Pentru a vizualiza update-urile recepționate de ruterul local se foloseşte comanda **show cdp neighbors.**

Capitolul 16 Protocoale pentru rutare

Există trei metode prin care un ruter *învață* traseul unui pachet către destinație:

- rute statice – sînt definite manual de către administratorul de rețea sub forma următorului *hop* (următorul ruter sau gateway) către destinație. Sînt folosite mai ales în rețelele mici cînd se doreşte reducerea traficului.
- rute implicite – sînt definite manual de către administratorul de rețea sub forma traseului pe care îl urmează un pachet atunci cînd nu se cunoaşte calea către destinație.
- rute dinamice – ruterul învață traseele către destinație prin recepționarea actualizărilor oferite de alte rutere din rețea.

16.1 Caracteristici ale protocoalelor pentru rutare

Identificarea tarseului pe care trebuie să îl urmeze un pachet este funcția ruterului prin care acesta evaluează traseelor către destinație şi care dintre acestea este cel mai bun. Rutarea se referă la procesul prin care se alege cel mai bun traseu pentru transmiterea pachetelor către destinație.

Toate protocoalele pentru rutare au acelaşi obiectiv: partajarea informațiilor despre rețea între ruterele participante. Acest obiectiv este însă atins în moduri diferite. Unele protocoale tarnsmit tabele de rutare între rutere în timp ce altele trasmit informații doar despre conexiunile directe pe care le gestionează.

Deosebirile nu se limitează numai la cele menționate anterior. Caracteristicile cu privire la perofmanțe şi sclabilitate sînt diferite de la un protocol la altul.

Protocoale distance-vector

În categoria protocoalelor distance-vector (şi derivatele acestora) sînt incluse:

- **RIP- Routing Information Protocol, versiunile 1 şi 2**
- **IGRP – Interior Gateway Routing Protocol**
- IPX RIP – NetWare Internetwork Packet Exchange Routing Information Protocol

- RTMP – AppleTalk Routing Table Maintenance Protocol
- AURP – AppleTalk Update-Based Routing Protocol
- **EIGRP – Enhanced IGRP**
- **BGP – Border Gateway Protocol**

Ce înseamnă de fapt vector-distance? Termenul se referă la direcția (şi informațiile aferente) pe care trebuie să o parcurgă pachetele pînă la destinație, dirrecție exprimată de cele mai multe ori sub forma de *hop-count uri.* Hop count reprezintă numărul ruterelor care trebuie traversate de un pachet pentru a ajunge la rețeaua destinație (pentru unele protocoale această unitate de măsură se referă la numărul legăturilor).

Un protocol distance vector întreține şi transmite tabele de rutare în care sînt listate toate rețelele cunoscute şi distanțele către fiecare din acestea. O tabelă de rutare arată cam aşa:

Rețea	**Distanța în *hopuri***	**Hopul următor**
10.0.0.0	0 (conexiune directă)	Interfața 1
172.16.0.0.	0 (conexiune directă)	Interfața 2
172.17.0.0.	1	172.16.0.2.
172.18.0.0.	2	172.16.0.2.
192.168.1.0	1	10.0.0.2
192.168.2.0	2	10.0.0.2

Protocolul pentru rutare transmite o astfel de tabelă tuturor ruterelor către care există o conexiune directă. Tabela de rutare îmbracă forma unui packet broadcast care este transmis la anumite intervale de timp. Dacă protocolul de rutare suportă tehnica denumită *split horizon,* ruterul transmite doar rutele care pot fi atinse. Această tehnică reduce intervalul la care trebuie fpăcută actualizarea tabelelor de rutare şi creşte acurrateţera informațiilor despre rutare. Mai mult, informațiile care pot gestionate mai bine local de către un ruter nu vor fi transmise celorlalte rutere.

O altă tehnică implementată de aceste protocoale se referă la *hold down timer*: maniera standard prin care ruterul încearcă să rezolve problema buclelor. Pentru a fi mai lcari vom exemplifica pe cele două rețele din imaginea următoare.

Rețeaua	Distanța	Următorul hop

172.16.0.0	0	Interfața 1
192.168.2.0	1	Ruter B

Rețeaua	Distanța	Următorul hop
192.168.2.0	0	Interfața 1
172.16.0.0	1	Ruter A

Cînd ruterul îşi transmite tabela de rutare, el distribuie doar informațiile cu privire la rețeaua şi distanța pînă la acea rețea nu şi următorul hop. Acesta este motivul pentru care apar buclele. Secvența unui eveniment care poate conduce la o rutare în buclă este cam aşa:

1. Conexiunea ruterului A la rețeaua 172.16.0.0 pică
2. Ruterul A şterge din tabela de rutare informațiile cu privire la rețeaua 172.16.0.0
3. În baza informațiilor precedente obținute de la ruterul A, ruterul B îşi transmite tabela de rutare anunțînd că poate găsi rețeaua 172.16.0.0
4. Ruterul A adaugă în tabela de rutare rețeaua 172.16.0.0 cu distanța 2 şi următorul hop ruterul B
5. Ruterul A primeşte un cadru de la un calculator din rețeaua 172.16.0.0
6. Ruterul A transmite cadrul ruterului B
7. Ruterul B transmite cadrul ruterului A
8. şi tot aşa până cînd valoarea TTL din cadru expiră

Dacă nu ar exista mecanismul split horizon problemele se complică şi mai mult. La un moment dat ruterul A va transmite o actualizare a rutelor sale prin care anunță că poate ajunge în rețeaua 172.16.0.0. Ca urmare, ruterul B va actualiza ruta respectivă cu o distanță egală cu 3. Ambele rutere vor continua să-şi trimită actualizări pînă cînd cîmpul distnță ajunge la o valoare infinită (16 în cazul RIP-ului). În acest moment ruterul şterge respectiva rută din tabelă

Problema actualizării informațiilor de către ruter este cunoscută sub denumirea *count-to-infinity*. Funcția hold down este cea care spune ruterului să nu adauge în tabela de rutare informații despre o rută care a fost modificată pînă cînd nu expiră timpul pentru hold-down.

În cazul nostru, dacă ruterul A face apel la această funcție, nu va adăuga ruta pentru rețeaua 172.16.0.0. trimisă de ruteruul B. Acest lucru se întîmplă şi dacă ruterul B face apel la split horizon, pentru că nu va transmite ruterului A informații despre rețeaua 172.16.0.0.

Protocoale link state

Această categorie de protocoale pentru rutare nu schimbă între ele tabele de rutare sub forma prezentată anterior. Informațiile transmise de rutere vizează starea legăturilor (rețelelor) conectate direct. Ruterul care foloseşte un astfel de protocol transmite periodic un pachet multicast prin care oferă informații despre starea conexiunilor sale. Ruterele care primesc acest pachet îl vor transmite mai departe ruterelor conectate direct.

Se includ în această categorie următoarele protocoale:

- OSPF – Open Shortest Path First
- IS-IS – Intermediate System – to – Intermediate – System
- NLSP – NetWare Link Services Protocol.

În general convergența realizată de aceste protocoale este mult mai rapidă față de protocoalele distance-vector dar acest avantaj are în spate puterea crescută a procesoarelor şi memoriei de care trebuie să dispună ruterul. La cele spuse anteriro se mai adaugă uşurința cu care poate fi administrat un protocol distance-vector față de unul link-state..

Unităţi de măsură, ierarhii şii convergenţă

Pentru a determina ruta pe care trebuie transmis un pachte, protocoalele de rutare fac apel la diferite unităţi de măsură. În mod obişnuit, protocoalele de tip distance vector folosesc aşa după cum am pomenit deja, *hop count*-ul. Protocoalele mai noi pot calucla rutele ţinând cont şi de alte caracteristici ale reţelei: întîrzierea, lăţimea de bandă disponibilă etc. Scalabilitatea unei reţele este direct afectată de unităţile de măsură folosite. De exemplu, un protocol precum RIP-ul poate să nu selecteze o rută cu o lăţime de bandă mai bună dar care depăşeşte limita de 15 hop counturi.

Chiar dacă toate ruterele îndeplinesc aceleaşi sarcini, anumite protocoale nu suportă ierarhii. Protocoalele dinaceastă categorie grupează ruterele în zone de lucru, sisteme autonome sau domenii de lucru. Într-o astfel de organizare, unele rutere comunică cu cele din zona locală în timp ce altele au ca sarcină interconectarea domeniilor sau sistemelor autonome.

Dacă privim protocoalele prin prisma locaţiei în care sînt folosite, le putem împărţi în două categorii: de interior şi de exterior.

Protocoalele de interior (RIP, OSPF, IGRP) sînt folosite de ruterele din cadrul aceleaşi organizaţii sau sistem autonom (sistem autonom = grup de rutere aflat sub o administrare centralizată). La polul opus, protocoalele de exterior (BGP) rutează între diferite sisteme autonome.

O altă caracteristică a protocoalelor de rutare vizează maniera de tratare a adreselor IP. Protocoalele de rutare de tip *classful* (RIP v.1, IGRP) calculează lungimea prefixului (porţiunea netork) unei adrese pentru a determina din ce clasă face parte. Un astfel de protocol nu transmite nici o informaţie cu privire la lungimea prefixului calculat. Notaţia care se foloseşete tot mai des în acest caz este următoarea : 10.1.0.1/16. Acest lucru înseamnă că prefixul reţelei are o lungime de 16 biti ceea ce însemnă o mască de reţea cu valoarea 255.255.0.0.

Protocoalele de rutare de tip *classless* (RIP v.2, EIGRP, OSPF, BGP, IS-IS) transmit lungimea prefixului împreună cu cu adresa IP. În această manieră, reţelele pot fi grupate sub forma unei singure intrări în tabela de rutare folosind lungimea prefixului pentru a specifica care reţele au fost grupate.

Convergenţa se referă la timpul necesar unui ruter pentru a înţelege de o manieră consistentă topologia unei reţele atunci cînd au avut loc schimbări în cadrul acesteia. Prin schimbări în cadrul unei reţele vom înţelege segmentări, defectarea unui ruter, adăugarea unui ruter etc. Convergenţa este o componentă critică în cadrul proiectării unei reţele.

Acest proces este iniţiat cînd ruterul este notificat că o legătură către unul din ruterele pereche devine inactivă (de exemplu). Un ruter de la CISCO transmite cadre *keepalive* la fiecare 10 secunde pentru a determina starea unei legături.

Într-o reţea WAN, ruterul transmite aceste cadre către ruterul aflat la capătul unei legături în timp ce într-o reţea LAN aceste cadre sînt transmise către o adresă multicast

16.2 RIP – Routing Information Protocol

RIP a fost primul protocol standardizat dezvoltat pentru mediile TCP/IP. După cum aminteam şi mai devreme, acesta este un protocol de tip *distance vector* destul de uşor de utilizat şi administrat.

Acest protocol transmite broadcast tabnla sa de rutare la fiecare 30 de secunde. Unpachet poate să conţină pînă la 25 de rute, ceea ce pentru reţele mari reprezintă un

dezavantaj. Pentru rețele care vor folosi acest protocol trebuie ținut cont de faptul că folosește hop count-ul ca unitate de măsură (maxim 15 rutere) chiar dacă alte rute au o lățime de bandă mai bună, congestie redusă etc.

IETF a dezvoltat cea de a doua versiune a acestui protocol prin care unele din problemele legate de scalabilitate au fost rezolvate. RIP versiunea 2 adaugă următoarele cîmpuri intrărilor din tabela de rutare:

- **route tag –** un indicator care face diferența între rutele interne din cadrul domeniului RIP și rutele externe care au fost importate de la alte protocoale de rutare sau de la sisteme autonome diferite.
- **Subnet mask –** conține masca de rețea care va fi aplicată prefixului din cadrul adresei IP.
- **Next hop –** specifică adresa IP a următorului ruter către care va fi transmis pachetul.

Cîmpul *route tag* este cel care unifomizează rețele ce folosesc RIP cu cele non-RIP. Faptulcă în versiunea 2 a fost adăugat și cîmpul subnet mask oferă suport pentru rutarea de tip classless. Mai multe detalii în RFC 1723.

16.3 IGRP – Interior Gateway Routing Protocol

IGRP este un protocol dezvoltat de CISCO la mijlocul anilor 80. Multe din rețele au început să folosească acest protocol pentru că depășește limitările la care erau supuse de către RIP. IGRP efectuează actualizarea tabelelor de rutare la fiecare 90de secunde fapt care conduce la o utilizare mai eficientă a lățimii de bandă.

În calcularea rutelor disponibile, IGRP ia în calcul următorii factori:

- Lățimea de bandă – administratorul poate configura lățimea de bandă disponibilă pentru un segment de rețea sau o poate folosi pe cea implictă care este specifică tipului legăturii existente
- Întîrzierea – nu este calculată în mod dinamic. Se are in vedere suma întîrzierilor de pe fiecare interfața de ieșire din cadrul unei rute.
- Încrederea – este calculată dinamic în funcție de posibilitatea transmiterii și recepționării pachetelor de tip keepalive. Trebuie configurată prin intermediul comenzii *metric weights.*
- Încărcătura – nu este folosită dacă nu este configurată comanda precedentă. În acest caz gradul de încărcare al rețelei este calculat în mod dinamic.

Spre deosebire de RIP, protocolul IGRP deține un algoritm mai eficient prin care selectează ruta implicită. Cînd se folosește RIP administratorul stabilește o singură rută implicită care este identificată ca fiind rețeaua 0.0.0.0.

IGRP-ul permite ca rețelele *reale* să fie marcate ca fiind candidate la rute implicite. Periodic are loc o scanare a acestor rute candidate și se alege cea acre are cele mai bune valori ale unităților de măsură pentru a deveni rută implicită.

16.4 OSPF – Open Shortest Path First[29]

[29] Definit prin RFC 2178

La începutul anilor 80, IETF recunoştea nevoia existenței unui protocol de interior care să depăşească limitările impuse de RIP. Rezultatul muncii IETF s-a concretizat în apariția OSPF.

Dintre avantajele acestui protocl de rutare amintim:

- Este un standard deschis acceptat de majoritatea producătoriulor de echipamente
- Convergența este mai rapidă decît la RIP
- Oferă o securitate mai bună
- Oferă suport pentru VLSM (variable-length subnet mask)
- Transmite cadre multicast în loc de broadcast
- Nu consumă prea mult din lățimea de bandă a rețelei

Pentru a diminua utilizarea lățimii de bandă OSPF transmite în rețea doar schimbările intervenite în cadrul tabelelor de rutare. Traficul în rețea se limitează la sincronizarea bazelor de date, sincronizare ce apare la fiecare 30 de minute.

Ruterele care rulează acest protocol acumulează informații cu privirea la starea legăturilor existente şi calulează cel mai scurt drum către o anumită rețea. Acest algoritm de calcul se numeşte SPF (shortest patj first) sau Dijkstra. Rezultatul acestui calcul este o bază de date cu topologia cunoscută (link state database).

Din această bază de date, fiecare ruter construieşte un arbore cu cele mai scurte rute, arbore în care ruterul care realizează acest calcul este rădăcina.

16.5 CIDR – Classless Inter-Domain Routing

La începutul anilor 90 IETF anunța pe un ton sumbru că în martie 1994 adresele IP de clasă B vor fi epuizate. În absența unui nou mecanism de adresare, sclalabilitatea Internetului urma să fie compromisă. Singura soluție viabilă pe termen lung o reprezenta crearea unui nou IP cu un spațiu de adresare mult mai larg: *IPng – Internet Protocol Next Generation* sau Ipv6 (adresare pe 128 biti). Pe termen scurt neajunsurile au fost eliminate prin adoptarea unei arhitecturi mai flexibile: CIDR – Classless Inter-Domain Routing

Pînă la IPng , la neajunsurile ridicate de reducerea adreselor IP şi de saturarea tabelelor de rutare au apărut şi alte soluții: Variable Length Subnet Masking (VLSM) – numai în cazul anumitor prtocoale de rutare, Address Allocation for Private Internets, Network Address Translation, CIDR.

Schema folosită de CIDR nu se deosebeşte prea mult de Ipv4 fiind de fapt o extensie a acestuia. Flexibilitatea noii scheme este dată de biții care identifică cele trei clase principale de adrese, cu un prefix de rețea. Conceptul de clasă dispare fiind înlocuit cu cel de *prefix*.

Superneting-ul presupune de fapt combinarea a două sau mai multe adrese de rețea consecutive.

Să luăm ca exemplu o adresă din fosta clasă C: 192.168.8.0. Dacă la această adresă se aplică prefixul /22 înseamnă că rămîn 10 biti ce pot fi afectați hosturilor. O adresă de clasă C nu poate însă avea 1022 hosturi. Vom putea de exemplu forma patru rețele de clasă C pornind de la 192.168.8.0/22. Ruterul va înțelege că încă trei rețele sînt ataşate la acesta şi orice pachet către 192.168.9.0, 192.168.10.0 sau 192.168.11.0 va fi rutat cu referire la 192.168.8.0/22. CIDR nu funcționează decît cu blocuri de adrese IP continue.

Alocarea adrselor în astfel de blocuri conduce la ceea ce se numeşte rutare ierarhică sau agreagrea rutelor: o singură rută de nivel superior poate reprezenta mai multe rute de nivel inferior în tabelele de rutare. Schema după care se face rutarea este similară sistemului

telefonic: centrala apelantă analizează doar prefixul apelant după care rutează apelul către respectiva centrală.

Cum se face agregarea rețelelor?

Rețeaua	**În binar**
192.168.8.0	11000000.10101000.00001000.00000000
192.168.9.0	11000000.10101000.00001001.00000000
192.168.10.0	11000000.10101000.00001010.00000000
192.168.11.0	11000000.10101000.00001011.00000000
SM: 255.255.252.0	11111111.11111111.11111100.00000000

După cum se vede din tabelul de mai sus, cel de al şaselea bit din octetul trei al adreselor de rețea este identic. Urmînd această regulă, toate hosturile acestor rețele vor face parte dintr-o singură rețea.

Să presupunem ănsă că dorim să facem superneting pornind cu rețeaua 192.168.10.0/22. Înseamnă că vom avea încă trei rețele: 192.168.11.0, 192.168.12.0, 192.168.13.0. Transformînd în binar aceste adrese vom observa că cel de al şaselea bit al celui de al treilea octet nu este identic. Rețele 192.168.10.0 şi 192.168.11.0 vor face parte dintr-o altă superrețea.

Rețeaua	**În binar**
192.168.10.0	11000000.10101000.00001010.00000000
192.168.11.0	11000000.10101000.00001011.00000000
192.168.12.0	11000000.10101000.00001100.00000000
192.168.13.0	11000000.10101000.00001101.00000000
SM: 255.255.252.0	11111111.11111111.11111100.00000000

Pentru a afla cîte rețele au fost agregate vom scădea din valorea celui de al treilea octet valorea subnet maskului: 256-252.

Capitolul 17. Proiectarea rețelelor

17.1 Câteva considerații

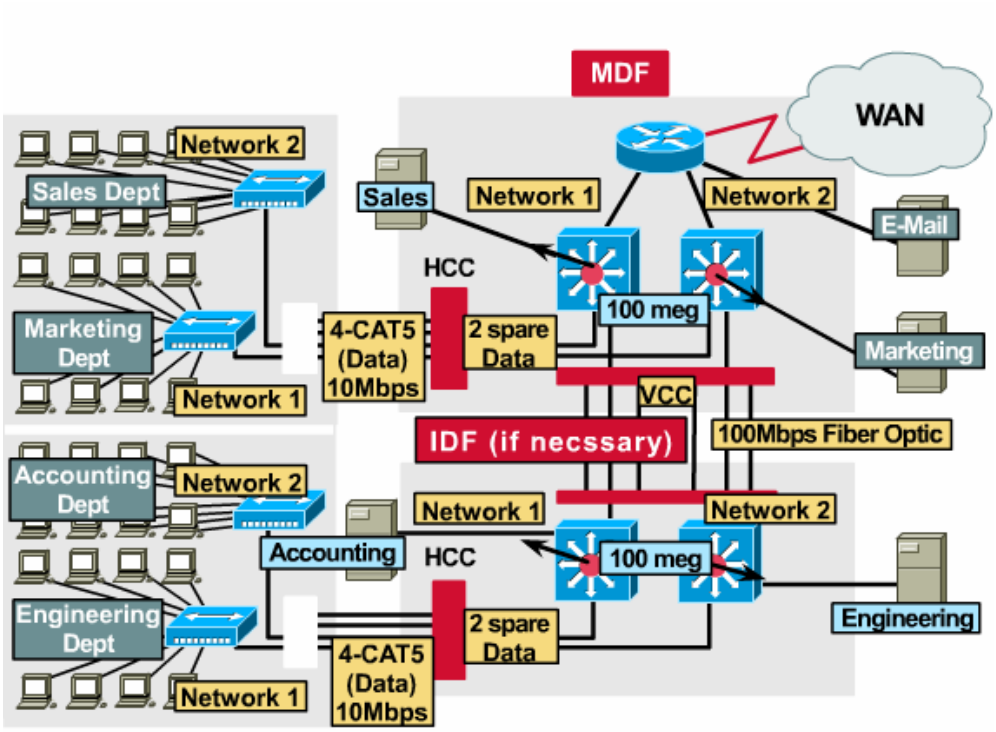
Proiectarea rețelelor înseamnă întotdeauna mai mult decât conectarea a două sau mai multe calculatoare între ele. Prima etapă a oricărui astfel de proiect constă în stabilirea unor obiective, specifice fiecărei organizații în parte. Chiar dacă avem în vedere specificul firmei, când proiectă o rețea trebuie să ținem cont de:

- **Funcționalitate:** rețeaua trebuie să funcționeze optim
- **Scalabilitate:** rețeaua trebuie să ofere posibilități de dezvoltare ulterioară.
- **Adaptabilitate:** rețeaua trebuie dezvoltată astfel încât să nu includă elemente care îi vor restricționa dezvoltările ulterioare.
- **Gestionare:** rețeaua trebuie dezvoltată astfel încât să permită monitorizarea.

Dezvoltările tehnologice ale ultimilor ani impun în multe cazuri reproiectarea rețelelor actuale. Avem în vedere în primul aplicațiile multimedia și tehnologiile high speed. Este motivul pentru care atunci când pornim un astfel de proiect trebuie să avem în vedere:

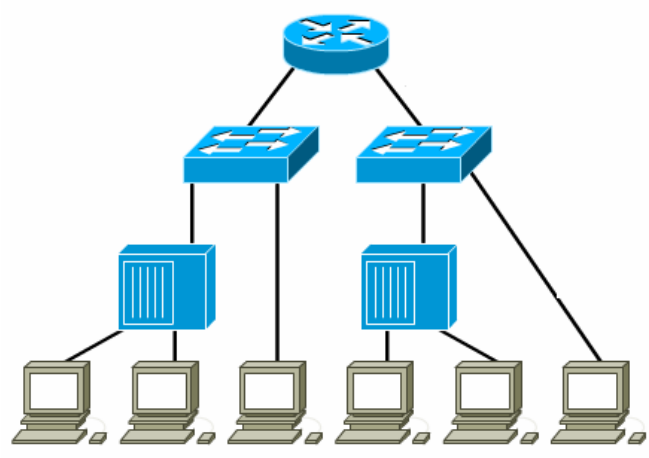
- Funcția și amplasamentul serverelor
- Detectarea coliziunilor
- Segmentarea
- Lățimea de bandă și domeniile de broadcast.

Primul element care conduce la succesul unei rețele îl reprezintă înțelegerea funcției pe care o îndeplinesc serverele și amplasarea acestora. Dacă e să facem o clasificarea a serverelor pe care le întâlnim într-o firmă putem vorbi de servere organizaționale și servere pentru grupuri de lucru. Serverele organizaționale ar trebui să fie cele care oferă servicii tuturor utilizatorilor firmei: e-mail, www.



Serverele organizaţionale trebuie amplasate în MDF (Main Distribution Facility) pentru a se evita ca traficul către acestea să traverseze şi celelalte reţele. Ideal ar fi ca serverele grupurilor de lucru să fie amplasate în IDF (Intermediate Distribuiton Facility), sau altfel spus cît mai aproape de utilizatorii care au nevoie de aplicaţiile găzduite de aceste servere. În acest mod traficul informaţional va traversa infrastructura IDFului respectiv fără a mai afecta şi alţi utilizatori. Switchurile dintre MDf şi IDF ar trebui să aibă 100Mbps sau chiar mai mult. Selecţia şi amplasarea echipamentelor este o decizie care trebuie să aibă la bază în primul rând reducerea coliziunilor în reţea. Numărul broadcasturilor devine excesiv în momentul în care există prea multe pachete-client ce necesită servicii, sau prea multe pachete-server care anunţă servicii, actualizări ale tabelelor ARP etc.

Pe măsură ce reţeaua se dezvoltă incluzând din ce în ce mai multe noduri pe acelaşi segment, şansa cu un astfel de nod să nu mai funcţioneze este destul de mare: creşte numărul coliziunilor. Chiar dacă acest fenomen este ceva normal în cazul Ethernet, un număr excesiv de coliziuni contribuie la reducerea lăţimi de bandă. În majoritatea cazurilor, lăţimea de bandă teoretică se reduce cu până la 40% datorită coliziunilor. Soluţia o reprezintă bineînţeles segmentarea: divizarea unui domeniu de coliziune în mai multe astfel de domenii.



Prin folosirea switchurilor într-o rețea cu o topologie logică bus pot fi create mai multe domenii de coliziune, chiar dacă domeniul de broadcast nu va fi redistribuit (figura anterioară). Pachetele broadcast vor fi vizibile pentru toate stațiile domeniului. Prin urmare putem spune că scalabilitatea lățimii de bandă este dependentă de traficul total din rețea, iar scalabilitatea unui domeniu de broadcast este dependentă de totalul traficului broadcast care are loc.

Întregul trafic asociat portului unui switch Ethernet poate fi numit *domeniul lățimii de bandă* sau *domeniu de coliziune.* Toate stațiile de lucru aparținând unui astfel de domeniu se concurează între ele în folosirea lățimii de bandă. Traficul pe care îl face oricare host din cadrul domeniului este vizibil celorlalte stații.

17.2 Identificarea şi analizarea cerințelor

Pentru ca o rețea să poată deservi cerințele utilizatorilor trebuie să fie proiectată şi implementată respectând câteva etape:

- Identificarea cerințelor utilizatorilor;
- Analiza acestor cerințe;
- Proiectarea topologiei rețelei (structura de nivel 1, 2 sau3);
- Documentarea proiectului şi implementarea.

Primele informații pe care trebuie să le avem în vedere când pornim un astfel de proiect se referă la structura organizațională a firmei. Aceste informații includ: istoricul firmei şi situația actuală, dezvoltarea viitoare, politici operaționale şi proceduri manageriale, punctul de vedere al celor care vor lucra în condițiile implementării rețelei.

În această etapă trebuie să se obțină răspunsuri la întrebări de genul: care sînt angajații ce vor folosi rețeaua? Care este nivelul lor de cunoştințe din acest domeniu? Ce atitudine adoptă în legătură cu noile tehnologii? Aceste răspunsuri vă vor ajuta în identificarea numărului de angajați de care va fi nevoie pentru funcționarea rețelei în bune condiții şi a nivelului lor de pregătire.

Ideal ar fi ca informațiile obținute în această etapă să fie de ajuns pentru a identifica problema cît mai exact şi pentru a determina care operațiuni din activitatea firmei sînt

considerate critice. În continuare trebuie identificată persoana din cadrul firmei care are autoritatea de a stabili schema de adresare ce va fi folosită, schema folosită în denumirea calculatoarelor, topologia ce se doreşte şi configurația echipamentelor. Există trei categorii de resurse care pot afecta implementarea unei rețele : hard/soft, umane şi financiare.

Nu trebuie să trecem cu vedere disponibilitatea rețelei care poate fi influențată de throughput, timpul de răspuns şi accesul al resurse. Fiecare potențial client are propria definiție a disponibilității. Este motivul pentru care trebuie analizate cerințele utilizatorilor. De cele mai multe ori rețelele se proiectează sau reproiectează ca urmare a creşterii cerințelor acestora.

După identificarea cerințelor generale ale rețelei, următoarea etapă constă în alegerea unei topologii care să satisfacă cerințele utilizatorilor. Configurația dominată pe piața actuală o reprezintă topologia star sau extended star, motiv pentru care de acestea ne vom ocupa în continuare.

Proiectarea unei topologii poate fi descompusă în trei categorii principale corespunzătoare primelor trei niveluri ale modelului OSI.

Când vorbim de medii de comunicații ne referim la nivelul fizic al modelului OSI. Această zonă este acoperită prin standardele ANSI/TIA/EIA 568 A cu privire la cablarea structurată.

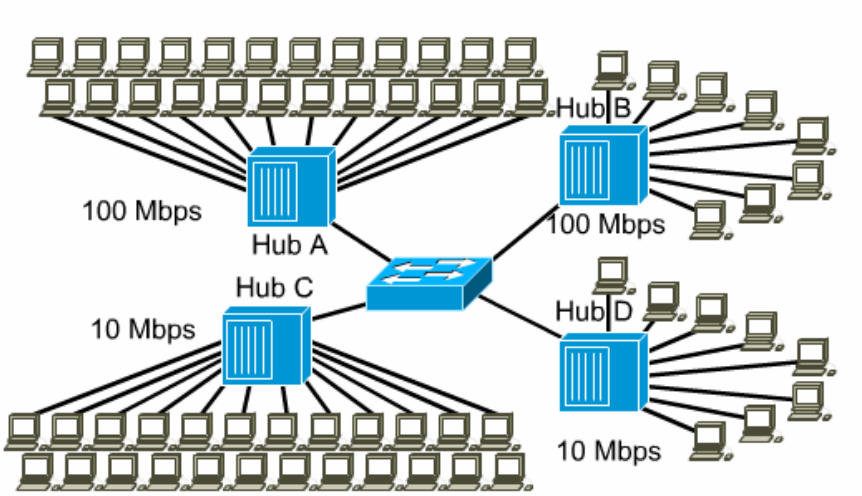
Standardele EIA/TIA prezintă şi care este nivelul atenuării şi al încrucişării indus de fiecare tip de mediu de transmisie.

Imaginea anterioară se numeşte diagramă logică de rețea. O astfel de diagramă include:

- Localizarea exactă a MDF, IDF şi a dispozitivelor folosite în cadrul acestora
- Tipul şi cantitatea de cablu folosite pentru a interconecta IDF de MDF.
- Documentarea cablurilor: etichetare, portul de intrare (HC şi VC).
- Posibilitățile de upgrade.

Prin instalarea de switchuri în MDF şi IDF şi a cablurilor verticale între acestea, apărea nevoia creşterii capacității acestora legături comparativ cu restul rețelei. Ştim că pentru cablarea orizontală se foloseşte UTP cat 5 cu o lungime maximă de 100m. Pentru majoritatea rețelelor, o lățime de bandă de 10Mbps este suficientă.

Ca urmare însă a folosirii switchurilor asimetrice, trebuie identificat numărul porturilor de 10 şi 100Mbps de care este nevoie în MDF şi fiecare IDF. Tot în acest moment trebuie identificată şi mărimea domeniului de coliziune prin stabilirea exactă a numărului de calculatoare ce se vor conecta la un port al switchului. Acest lucru va afecta şi lățimea de bandă disponibilă fiecărui host în parte. În practica curentă se folosesc hub-uri pentru a conecta mai multe hosturi la un singur port. Toate calculatoarele conectate la acelaşi hub vor partaja acelaşi domeniu de coliziune şi aceeaşi lățime de bandă.



La ora actuală există încă switchuri care nu întrețin tabele cu adrese MAC multiple (mapate pe fiecare port în parte). Prin urmare problema broadcastului nu este rezolvată în totalitate.

Soluția folosirii huburilor impune punerea de acord a specificațiilor obținute în faza de analiză cu problema domeniilor de coliziune şi a lățimii de bandă. Aceasta deoarece pe măsură ce rețeaua se dezvoltă şi cerințele legate de lățimea de bandă vor fi mai mari. În cazul cablării orizontale, trecerea de la 10 Mbps la 100 Mbps impune identificarea cerințelor legate de disponibilitatea porturilor pe 100.

Folosirea ruterelor are ca efect crearea de rețele unice din punct de vedere logic şi fizic. Am mai spus deja că cel mai important aspect al rețelelor îl reprezintă numărul broadcasturilor (cereri ARP de exemplu). Prin implementarea VLANurilor acest număr poate fi limitat. Comunicarea între VLAN-uri are loc însă prin intermediul ruterelor.

Când rețelele sînt împărțite în subrețele trebuie dezvoltată schema de adresare IP care va fi folosită. Cum ştim dacă trebuie să folosim rutere sau switchuri? Dacă problema ce se vrea a rezolva ține mai mult de un protocol decât de conexiune, atunci se recomandă folosirea ruterelor: broadcast în exces, securitate, adresare la nivelul rețea. Rețelele mari trebuie să includă obligatoriu rutere.

Capitolul 18. Network management

Prima şi cea mai importantă componentă care contribuie la realizarea unei reţele performante o reprezintă documentaţia acesteia. Din nefericire, în practică acesta este aspectul trecut cu vederea de cele mai multe ori. Documentaţia reprezintă, poetic spus, memoria oricărui administrator.

4.1 Documentaţia reţelei

Un astfel de document trebuie să includă în mod obligatoriu:

- Diagrame care să prezinte localizarea fizică a mediilor de transmisie
- Tipul cablurilor folosite şi caracteristicile acestora
- Lungimea fiecărui cablu
- Tipul terminatorilor folosiţi
- Schema de etichetare a fiecărui cablu

La acestea se mai adaugă planul fizic şi logic al MDF şi al IDFurilor: rackuri, echipamente, servere şi modalităţi de identificare. Pentru fiecare server şi staţie din reţea trebuie să fie precizate detalii privind configuraţia folosită. Informaţiile pe care trebuie să le conţină documentaţia reţelei sînt oarecum standardizate.

Configuraţia hardware a calculatorului				
Server sau Workstation:				
Locaţia fizică:				
Producător şi model:				
Serial #:				
Nr. Inv.				
Removable Drives:				
Producător	Litera	Capacitate	Intern/Externa	Nr. Bay intern
Fixed Drives:				
Producător	Litera	Capacitate	Intern/Externa	Nr. Bay intern

Memorie curentă/maximă:	Curentă:		Maximă:		
Cartele periferice:					
Producător:	Model:	Tip:	IRQ	DMA	Memory Addr.
Network Interface Cards:					
Producător:	Adresa nodului.	Model:	Lan Driver:	IRQ	DMA	Memory Addr.
Comentarii:					

Configuratia imprimantei			
Localizare fizică:			
Producător şi model			
Serial #:			
Nr. Inv.			
ID #:			
Memorie curentă/maximă:	Curentă	Maximă:	
Sertare hîrtie (bins)	Bin #1 tip hirtie	Bin #2 tip hirtie	Bin #3 tip hirtie

Configuratie:			
Serial			
Paralel			
Network	IP	Polling	MAC addr.
Operatori:			
Comentarii:			

Pentru fiecare calculator trebuie să existe o listă cu softul standard şi cel special instalat.#

Software					
Nr. Inv.:					
Sistem operare					
Producător	Versiune	Updates	Network	Securitate	
Application software					
Producător	Versiune	Updates	Network	Install directory	Data directory

De asemenea trebuie să existe o listă cu tyoate reparațiile efectuate asupra echipamentelor care fac parte din rețea.

Reparații calculator

Nr. Inv:			**Data**	
Tipul defecţiunii	Hardware:		Software:	
Descriere:				
Acoperire prin garanţie	Da	Nu	**Localizarea reparaţiei**	
Descrierea reparaţiei				
Departament/compartiment implicat				
Persoană autorizată				
Reparaţie efectuată de				
Comentarii				

Nu trebuie trecută cu vederea securitatea reţelei. Aceasta nu se referă doar la securitatea pe care o asigură softul (drrepturile utilizatorilor, parolele, firewall) ci şi securitatea fizică.

Securitate sală echipamente					
Localizare fizică				**Data:**	
Securitate fizică	Sistem închidere uşi	Ferestre	Tavane false	Detectoare incendiu	Detectoare mişcare
Tape Backup pentru servere					
Denumire server	Tip	Media	Off-site Loc.	Numele tape-set-ului	Ziua de început
Server 1					
Server 2					
Server 3					

Acces autorizat:	Nume	Departament	Functie
Comentarii			

Securitate utilizatori rețea				
Localizare fizică			**Data:**	
User name:			**User ID:**	
Departament **Lungime parolă:**			Dept. manager: Home Dir:	
Data expirării:			Acces local	
Interval orar acces			Acces imprimantă	
Acces de la distantă			Acces administrator	
Membru în grupuri	Numele grupului	Drepturi grup	Local/Global	Restricții
Sarcini de lucru/privilegii				
Comentarii				

Un astfel de document poate fi crucial pentru munca administratorului pentru că descrie modul în care utilizatorii interacționează cu rețeaua.
Informațiile conținute în documentele prezentate anterior alcătuiesc documentația rețelei. Întreținerea şi upgrade-ul devin mai simple dacă la baza lor există un astfel de document.

4.2 Securitatea rețelei

Când vorbim de securitate în zona rețelelor trebuie să avem în vedere două aspecte: protecția împotriva accesului neautorizat şi recuperarea datelor în urma unor evenimente neprevăzute.

Protecţia împotriva accesului neautorizat este asigurată prin politici de securitate şi prin intermediul sistemului de operare folosit, oricare ar fi acesta. În acest moment ne interesează mai mult modalităţile de recuperare a datelor sau copiile de siguranţă.

Din multitudinea de metode posibil de implementat ne vom ocupa doar de cele mai importante trei astfel de metode: backup-ul pe bandă, hard disk-uri tolerante la căderi şi folosirea UPS-urilor.

Backup pe bandă

Aceasta este procesul prin care se realizează duplicarea datelor de pe hard disk pe bandă magnetică. Motivul pentru care această tehnică este destul de mult folosită îl reprezintă costul redus şi capacitatea mare de stocare a datelor. Chiar dacă datele sînt scrise secvenţial acest lucru nu reprezintă o piedică în folosirea acestei tehnici.

Majoritatea tehnicilor folosite lucrează cu un *flag* sau *switch* numit *bit arhivă*. Un astfel de bit este memorat împreună cu fişierul şi activat ori de câte ori un fişier este creat sau activat. Acest flag indică în timpul backup-ului dacă fişierul necesită backup sau nu. Dacă fişierul este transferat pe bandă, acest bit este dezactivat indicând că pentru respectivul fişier a fost realizată copia de siguranţă.

Name	Size	Type	Modified	Archive Bit
App.tif	142KB	TIF File	1/20/94 10:11 AM	1
Altair.tif	966KB	TIF File	1/20/94 10:04 AM	1
Apple.tif	1,193KB	TIF File	1/20/94 9:28 AM	0
Printer.bmp	433KB	BMP File	1/18/94 1:19 PM	1
Diskclen.bmp	433KB	BMP File	1/18/94 12:00 PM	0
Misc.bmp	433KB	BMP File	1/17/94 6:50 PM	1
Plotter.bmp	433KB	BMP File	1/17/94 4:05 PM	1
Treestru.bmp	433KB	BMP File	1/17/94 11:18 AM	1
Rad_shk.bmp	433KB	BMP File	1/15/94 1:06 PM	0
Chip.bmp	433KB	BMP File	1/15/94 12:58 PM	1
Mac_pc.bmp	433KB	BMP File	1/15/94 12:48 PM	1
Ibm_pc.bmp	433KB	BMP File	1/15/94 12:45 PM	0

Există cinci metode de realizare a copiilor de siguranţă pe bandă magnetică:

- **Backup total**: toate fişierele de pe un hard disk sînt salvate şi pe bandă şi bitul arhivă are valoarea 0 (este off).
- **Backup incremental:** realizează copii ale tuturor fişierelor care au fost create sau modificate de la ultimul backup total. După cum observaţi, această tehnică nu poate fi folosită decât în conjuncţie cu un backup total. Orice fişier creat sau modificat are activat bitul arhivă (1=on) astfel încât să fie salvat în timpul următorului backup incremental
- **Backup diferenţiat:** seamănă cu backup incremental cu singura diferenţa ca bitul arhivă nu este resetat dacă un fişier este salvat pe bandă. Acest lucru înseamnă că de fiecare dată când are loc un astfel de backup, toate fişierele create sau modificate de la ultimul backup total vor trebui din nou salvate
- **Backup copie:** presupune copierea pe bandă numai a fişierelor indicate de utilizator/administrator. În acest caz bitul arhivă nu este resetat pe off.
- **Backup zilnic** presupune copierea numai a fişierelor care au fost modificate în timpul zilei în care se face backupul. Bitul arhivă nu este resetat pe off.

Să exemplificăm un backup incremental .(primele trei metode sunt şi cele mai folosite în practică). În prima zi a săptămânii trebuie realizat backup total al datelor de pe server. Acest lucru înseamnă că bitul arhivă al tuturor fişierelor va fi resetat. În următoarea zi, pe o bandă separata se va realiza un backup incremental. Acest lucru înseamnă că pe bandă vor fi înregistrate toate fişierele modificate în prima zi, bitul arhivă al acestora fiind resetat. Se continua în aceeaşi manieră pentru fiecare din următoarele zile. La începutul următoarei săptămâni procesul se reia. Deşi nu este consumatoare de timp, această metodă are un dezavantaj: refacerea datelor presupune restaurarea acestora de pe copia backup total şi apoi de pe fiecare backup incremental. Dacă o bandă se defectează

Pentru un backup diferenţiat, mai întâi trebuie realizat în prima zi a săptămânii un backup total. Astfel, biţii arhivă sînt resetaţi pentru toate fişierele salvate pe bandă. A doua zi, pe altă bandă se va realiza un nou backup diferenţiat salvându-se fişierele modificate în ziua respectivă. Biţii arhivă nu vor mai fi resetaţi. Pe această ultimă casetă cu bandă se continuă procesul pentru restul zilelor.

Principalul avantaj al acestei metode îl reprezintă folosirea numai a două benzi. Un dezavantaj ar fi că fişierele salvate cu o zi în urmă si nemodificate vor fi şi ele salvate împreună cu cele modificate. De asemenea, dacă banda care conţine un backup diferenţiat realizat vineri de exemplu, se defectează, se pierd informaţiile zilelor precedente.

Trebuie să mai facem câteva menţiuni. Un backup total făcut zilnic necesită o singură bandă pentru a putea reface starea iniţială a datelor. Este însă o acţiune mare consumatoare de timp. De multe ori, informaţiile cărora li se realizează copii de siguranţă nu se afla doar pe un server ci şi pe staţii de lucru. Aceste informaţii pot fi mai importante decât cele de pe servere. Metoda prin care se realizează copii de siguranţă pentru datele de pe staţiile de lucru depinde de situaţiile de la faţă locului.

Să presupunem că pe o staţie se lucrează cu o cantitate mare de informaţii (se ţine evidenţa contabilă a firmei). Aceste informaţii sînt folosite numai de utilizatorul care lucrează pe staţia respectivă. Responsabilitatea backup-ului revine utilizatorului acelei staţii. Realizarea copiei de siguranţă nu va afecta throughputul reţelei dacă staţia este dotată cu drive de bandă sau cu o unitate ZIP.

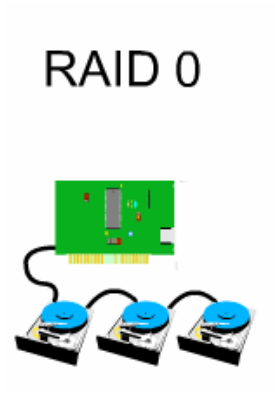
Dar dacă sînt mai mulţi utilizatori în această situaţie? Trebuie dotate toate staţiile cu dispozitive pentru backup? Dacă acest lucru este posibil, problema este ca şi rezolvată. Dacă nu, soluţia o reprezintă crearea pe server de directoare în care utilizatorii să-şi salveze datele. În acest fel, când se realizează backup-ul serverului sînt salvate şi informaţiile utilizatorilor.

În concluzie, o soluţie valabilă pentru orice situaţie nu există!

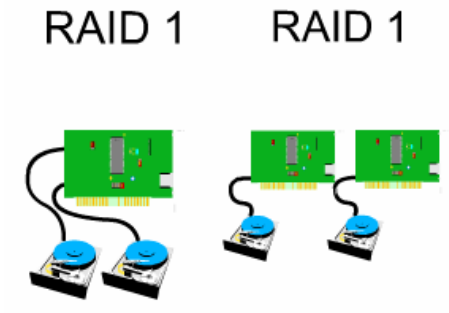
Redundanţa

Backup-ul nu este o tehnică ce se aplică singular. De cele mai multe ori ea lucrează în conjuncţie cu hard diskurile tolerante la căderi: **RAID** (Redundant Array of Inexpensive Disks) 0-5.

- *RAID 0* – datele sînt scrise pe mai multe discuri cu un singur controller, fără să se facă verificarea parităţii. Este folosit mai mult pentru creşterea ratei de transfer. Nu oferă posibilităţi de backup.



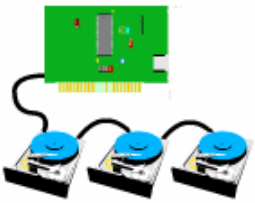
- *RAID 1* – se mai numeşte şi *disk mirroring* (discuri duble). Aceleaşi date sînt scrise pe două partiţii identice de pe două discuri diferite. Se pot folosi două controllere diferite pentru cele două hard diskuri. În cazul în care unul din discuri se defectează, datele vor fi recuperate de pe cel de al



doilea.
- *RAID 2* – Datele sînt scrise pe mai multe discuri fără a se face verificarea erorilor. Metoda nu se mai foloseşte deoarece necesită modificări substanţiale ale discurilor.
- *RAID 3* – Datele sînt scrise byte cu byte, un hard disk fiind folosit pentru verificarea parităţii. Este o metodă care nu se mai foloseşte.
- *RAID 4* – Datele sînt scrise sector cu sector. Un disc este folosit pentru verificarea parităţii. Viteza de scriere este destul de mică, motiv pentru care nici această soluţie nu mai este folosită.
- *RAID 5* – Datele şi informaţiile pentru verificarea parităţii sînt scrise simultan pe mai multe discuri (cel puţin 3). În cazul W2K, partiţia sistem nu poate face parte din sistemul RAID. Prin urmare în acest caz este nevoie de cel puţin 4 discuri. Mărimea partiţiilor trebuie să fie aceeaşi pe fiecare disc din setul RAID.

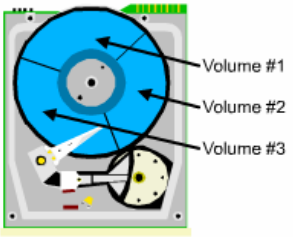
În cazul W2K, pentru a putea beneficia de un backup total se vor folosi 5 discuri: Primele două vor fi configurate ca RAID 1 pentru partiţia boot şi sistem, iar următoarele ca RAID 5.

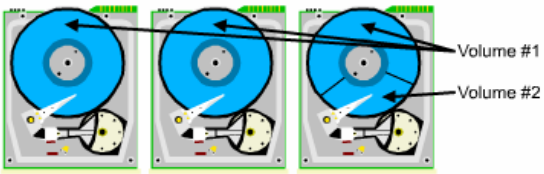
RAID 5



Nu putem încheia acest subiect fără a explica şi conceptul de *volume.* Acest termen se referă la o unitate fizică de memorare a datelor. Un set de volume de discuri concentrează spaţiul de mai multe discuri fără a oferi posibilitatea realizării backupurilor.

Seturile de volume sînt folosite doar pentru a ne putea referi la zone mari de memorie ca fiind o singură unitate.





Pe lîngă aceste aspecte trebuie avute în vedere problemele pe care le pot ridica interferenţele, descărcările electrostatice, condiţiile atmosferice şi, nu în ultimul rând viruşii.

4.3 Depanarea reţelei

Hotărât lucru nu putem să prevedem toate problemele care pot să apară într-o reţea. Chiar şi în această situaţie însă, depanarea nu trebuie să se facă haotic.

Să luăm exemplul unui utilizator care ne informează că nu mai poate accesa resursele Internetului. Din discuţii aflaţi că respectivul utilizator nu a făcut nimic în plus faţă de zilele anterioare. În schimb când analizaţi staţia pe care lucrează observaţi că a "suferit" un up grade cu o zi înainte. Prima dată vă duceţi cu gândul la driverul plăcii de reţea care s-ar putea să nu fie corect configurat.

Verificaţi acest lucru şi vă convingeţi că acest lucru a fost făcut corect. Verificaţi dacă serverul din respectiva subreţea răspunde la comanda **ping.** Nu răspunde. Verificaţi cablurile şi încercaţi încă o dată ping. Tot nu se conectează.

Pentru a fi siguri că nu sînt probleme între calculator, configuraţia driverului şi cartela de reţea se dă ping pe adresa de buclă a computerului: 127.0.0.1 (folosită pentru a ruta pachetele respectivului calculator prin placa de reţea)! Dacă se obţine răspuns, înseamnă că nu există astfel de probleme.

Vă gândiţi că s-ar putea să fie o problemă cu serverul. În acest caz încercaţi ping de pe o altă staţie din aceeaşi reţea. Surpriză: serverul răspunde la ping. Concluzia este că nu există probleme legate de conexiunea serverului.

Următorul pas presupune să vă duceți la IDF şi să schimbați portul switchului corespunzător stației şi să încercai din nou un ping. Ghinion, stația tot nu răspunde. Puneți la loc cablul şi mergeți în continuare pe fir. Adică faceți un nou patch cablu pentru stația respectivă.

Înlocuiți cablul şi încercați din nou ping. De această dată merge. Ca totul să fie făcut ca la carte nu uitați să documentați problema pe care tocmai ați rezolvat-o. Poate mai pățesc şi alții acelaşi lucru.